



Escola Politècnica Superior
d'Enginyeria de Vilanova i la Geltrú

UNIVERSITAT POLITÈCNICA DE CATALUNYA

TREBALL FINAL DE GRAU

TÍTOL: SISTEMA DE MONITORITZACIÓ I CARACTERITZACIÓ DEL
SISTEMA DE CAPTACIÓ D'ENERGIA D'UNA BOIA DE DERIVA
PER MESURA DE CORRENTS MARINS

AUTORS: Valls Vivas, Lluís

DATA DE PRESENTACIÓ: 05, 07 2019

RESUM

Aquest projecte es basa en dur a terme l'anàlisi i l'estudi d'un captador d'energia en una boia WAVY drifter. Aquesta, es una boia de deriva de petites dimensions amb un captador d'energia que permet alimentar-se de manera autosuficient amb els corrents marins i les ones produïdes pel mar parametritzant el seu flux a nivell oceànic.

En el captador, li hem introduït uns fotosensors per tal de poder veure a quina velocitat es capaç de girar, i per tant quina és l'energia que podem arribar a acumular.

Al llarg d'aquest projecte, realitzarem un estudi de les acceleracions que pateix la boia, i com aquestes, son capaces de moure el captador.

Mitjançant un controlador, decidirem com i en quin moment rebre i enviar les dades.

A través d'aquest controlador enviarem la informació captada per UDP, i serà mostrada per la pantalla d'un ordinador a temps real utilitzant el programa LabView.

Per poder obtenir tota aquesta informació utilitzarem un aparell anomenat TD1205P el qual es l'encarregat de captar les acceleracions de la boia i mitjançant el node ESP8266 captarem, processarem i enviarem aquesta informació a l'ordinador.

Al llarg d'aquest projecte es mostra la manera com s'ha actuat, les proves pertinents per saber quina es la millor configuració dels nostres aparells, i el posterior anàlisi.

En els apartats finals del projecte podrem veure la boia experimental que s'ha realitzat per dur a terme les proves al mar, juntament amb tots els elements introduïts per poder fer l'anàlisi, des de fotosensors fins al plom necessari per poder simular el pes que realment tindrà la boia. Aquesta haurà de romandre al mar per un període prolongat de manera autosuficient, és a dir, acumulant l'energia necessària per poder alimentar els sensors que té incorporats i enviar aquesta informació per tal de saber en tot moment la situació en que es troba a mesura que passa el temps.

Finalment es llençarà la boia al mar i es mostraran les conclusions pertinents per determinar si el projecte ha assolit els objectius els quals es va fixar a l'inici i veure si l'energy harvesting és capaç de captar l'energia suficient per poder auto alimentar-se.

Paraules clau (màxim 10):

Captador d'energia	Autoalimentació	Wavy drifter	Sensors autònoms
--------------------	-----------------	--------------	------------------

ABSTRACT

This project is based on carrying out the analysis and study of an energy harvesting in a wavy drifter buoy. This is a drift buoy of small dimensions with an energy sensor that allows self-sufficient feeding with marine currents and waves produced by the sea, parameterizing its flow at ocean level.

In the collector, we have introduced some photosensors to see what speed is able to rotate, and therefore what energy we can accumulate.

Throughout this project, we will carry out a study of the accelerations suffered by the buoy, and how they are able to move the sensor.

By studying this movement, we will be able to see if the sensor can feed all the sensors it contains and for how long.

Through a controller, we will decide how and when to receive and send the data.

Through this controller we will send the information captured by UDP, and it will be shown by a computer screen in real time using the LabView program.

In order to obtain all this information we will use a device called TD1205P which is in charge of capturing the buoy's accelerations and through the ESP8266 node we will capture, process and send this information in the computer.

Throughout this project we show the way we acted, the relevant tests to know the best configuration of our devices, and the subsequent analysis.

In the final sections of the project we can see the experimental buoy that was made to carry out the tests at sea, along with all the elements introduced to make the analysis, from photosensors to the necessary lead, to simulate the weight that really will have the buoy. This must remain at sea for a prolonged period in a self-sufficient manner, that is means, accumulating the necessary energy to feed the sensors that are incorporated and send this information in order to know at all times the situation in which it is as time passes .

Finally the buoy will be launched at sea and the relevant conclusions will be shown to determine if the project has reached the objectives that were set at the beginning and to see if the energy harvesting is able to capture enough energy to be able to self-feed

Keywords (10 maximum):

wavy drifter	self-sufficient	energy harvesting feeding	energy
--------------	-----------------	------------------------------	--------

Tabla de contenido

1. INTRODUCCIÓ.....	9
1.1 ESTAT DE L'ART: ENERGY HARVESTING	9
1.2 PROJECTE MELOA	9
1.3 MOTIVACIÓ	11
1.4 OBJECTIUS	12
1.5 ESTRUCTURA	13
2 ENERGY HARVESTING	14
2.1 DISSENY.....	14
2.2 FUNCIONAMENT	15
2.3 DADES ADQUIRIDES.....	16
3. PROPOSTA DEL DISSENY ACTUAL	18
3.1 MÈTODE OPERATIU	18
3.2 CANALS DE COMUNICACIÓ	19
3.2.1 COMUNICACIÓ SÈRIE.....	19
3.2.2 COMUNICACIÓ UDP.....	20
3.2.3 UBIDOTS	22
3.3 TRACTAMENT DE DADES.....	23
3.3.1 ACCELERACIÓ.....	23
3.3.2 TRANSFORMADA DE FOURIER.....	24
4 REQUISITS DEL DISSENY	27
4.1 WAVY DRIFTERS	27
4.2 TD1205P	28
4.2.1 PINOUT	28
4.2.2 ACCELERÒMETRE.....	29
4.2.3 COMANDES ATT PROGRAMADES	29
4.2.4 COMUNICACIÓ SIGFOX	30
4.3 NODE MCU ESP8266	30
4.3.1 GENERACIONS DEL NODE MCU	31
4.3.2 ESP-12E MÒDUL	33
4.3.3 ALIMENTACIÓ	34
4.3.4 ARRENCADA.....	34
4.3.5 METODOLOGIES DE FUNCIONAMENT DE BAIX CONSUM	35
4.3.6 SENSOR INFRAROIG	35
4.3.7 ADP5092 MAXIUM POWER POINT TRACKING	36
5 OBTENCIÓ DE VOLTATGE DE L'ENERGY HARVESTING	38
6. SOFTWARE DEL DISSENY.....	40
6.1 DIAGRAMES DE BLOCS.....	40
6.1.1 DIAGRAMA DE BLOCS NODE MCU ESP8862	40
6.1.2 DIAGRAMA DE BLOCS LABVIEW	42
6.2 LLIBRERIES UTILITZADES	47
6.3 ENTORN UBIDOTS.....	48
7 ANÀLISI ANTERIOR AL TEST.....	49
7.1 HARDWARE DEL ANÀLISI	49
7.2 TEST AMB EL NODE ESP8266 I EL TD1205P	50
7.4 TEST DE LA FREQUENCIA	53
8 BOIA PER DUR A TERME LES PROVES	55

9 METODOLOGIA EXPERIMENTAL.....	60
10 ANALISI DELS RESULTATS.....	62
11 AMPLIACIÓ AMB MULTIPLEXOR	65
12 DADES DEL PROJECTE	67
12.1 DESPESA ECONÒMICA	67
12.2 DURADA DEL PROJECTE	67
13 TREBALLS FUTURS.....	68
CONCLUSIONS	69

SUMARI DE FIGURES

- Figura 2.2. Dispositiu amb els dos cercles. El cercle intern amb rotació respecte a l'eix XC i l'extern amb rotació respecte a l'eix YC.
- Figura 2.3. Imatge frontal de l'arc interior amb els engranatges.
- Figura 2.4. Imatge dels rodaments one way.
- Figura 2.5. Imatges fetes durant la simulació.
- Figura 2.6. Gràfica de la potència instantània.
- Figura 3.1. Codi binari a enviar "97".
- Figura 3.2. Codi binari a enviar "97". [Font: Wikipedia].
- Figura 3.3. Acceleracions 2D. [Font: www.luisllamas.es].
- Figura 3.4. Acceleracions 3D. [Font: www.luisllamas.es].
- Figura 3.5. Trigonometria acceleracions 3D.
- Figura 3.6. Ona simple amb desfase. [Font: Wikipedia].
- Figura 3.7. Conjunt d'ones simples. [Font: www.conceptosclaros.com].
- Figura 3.8. Serie temporal i conjunt d'ones.
- Figura 3.9. Transformada de Fourier. [Font: www.conceptosclaros.com].
- Figura 4.1. Boia projecte MELOA. [Font: MELOA].
- Figura 4.2. TD1205P. [Font: Datasheet TD1205P].
- Figura 4.3. Pinout TD1205P. [Font: Datasheet TD1205P].
- Figura 4.4. Generacions ESP. [Font: Luis Llamas].
- Figura 4.5. Generacions ESP. [Font: Luis Llamas].
- Figura 4.6. Generacions ESP12E. [Font: Luis Llamas].
- Figura 4.7. Última Generació ESP. [Font: Luis Llamas].
- Figura 4.8. Pinout ESP. [Font: Datasheet ESP].
- Figura 4.9. Generacions ESP. [Font: yeoman.com].
- Figura 4.10. Generacions ESP. [Font: yeoman.com].
- Figura 4.11. Tipus d'arrencada ESP.
- Figura 4.12. Metodologia Baix consum.
- Figura 4.13. Configuració sensors infraroig [<https://hardwarehackingmx.files.wordpress.com>].
- Figura 4.14. Curves del Generador. [Font: Matias Garandell].
- Figura 5.1. Engranatge amb el fotosensor.
- Figura 5.2. Engranatges Energy Harvesting.
- Figura 5.3. Càlcul de la relació de transmissió.
- Figura 5.4. Rendiment del AD5092. [Font: Matias Garandell].
- Figura 6.1. Diagrama de blocs ESP part 1.
- Figura 6.2. Diagrama de blocs ESP part 2.
- Figura 6.3. Diagrama de blocs Labview part 1.
- Figura 6.4. Diagrama de blocs Labview part 2.
- Figura 6.5. Diagrama de blocs Labview part 3.
- Figura 6.6. Diagrama de blocs Labview part 4.
- Figura 6.7. Diagrama de blocs Labview part 5.
- Figura 6.8. Display Labview.
- Figura 7.1. Hardware proves.
- Figura 7.2. Connexió cable TTL. [Font: Datasheet TTL 232R].
- Figura 7.3. Informació port virtual.
- Figura 7.4. Labview TD1205P horitzontal.
- Figura 7.5. TeraTerm horitzontal.
- Figura 7.6. Labview 90 graus horitzontal.
- Figura 7.7. Labview 90 graus horitzontal.
- Figura 7.8. TeraTerm 90 graus horitzontal.
- Figura 7.9. Acceleració Eix Z 2G.
- Figura 7.10. Acceleració Eix Z 4G.
- Figura 7.11. Acceleració Eix Z 8G.
- Figura 7.12. Acceleració Eix Z 16G.
- Figura 7.13. Taula resum acceleracions.
- Figura 7.14. Resultat freqüència 1 Hz.
- Figura 7.15. Resultat freqüència 10 Hz.
- Figura 7.16. Resultat freqüència 25 Hz.
- Figura 7.17. Resultat freqüència 50 Hz.
- Figura 7.18. Resultat gràfica freqüència 50 Hz.
- Figura 8.1. Bola amb que simularem la boia.
- Figura 8.2. Espumant la boia.

Figura 8.3.Pesos introduïts a la boia.

Figura 8.4.Boia espumada amb pesos introduïts.

Figura 8.5.Fixació TD1205P.

Figura 8.6.Connectors.

Figura 8.7.Boia amb els connectors i els aparells pertinents.

Figura 8.8.Energy Harvesting.

Figura 8.9.Boia per fer el test.

Figura 8.10.Resultats abans de la prova final, fotosensor.

Figura 8.11.Resultats abans de la prova final, condensador.

Figura 9.1.Router i ordinador amb els que es van fer el test al mar.

Figura 9.2.Router i ordinador amb els que es van fer el test al mar.

Figura 9.3.Test de la boia al mar.

Figura 12.1.Multiplexor CD74HC4067 [<https://leantec.es/tienda/multiplexor-16-canales-cd74hc4067-analog-digital-multiplexer-mux-demux/>].

Figura 12.1.Taula de la veritat multiplexor [datasheet CD74HC4067]

1. INTRODUCCIÓ

1.1 ESTAT DE L'ART: ENERGY HARVESTING

Energy Harvesting significa el procés de captura i acumulació d'energia a mesura que aquesta va estant disponible, acumulant-la durant un període de temps i transformar-la per poder ser utilitzada mes endavant.

Aquesta energia pot esser emmagatzemada per una gran varietat de fons considerades despercebudes o inutilitzables per qualsevol motiu pràctic. Aquest procés també és conegut com captació d'energia, i per tant, és considerada una captació de l'energia lliure.

Aquesta energia lliure, es troba en el medi ambient com a residu d'altres processos. Com a exemples podríem trobar la vibració, estrès de la tensió, energia tèrmica, o mitjançant l'energia mecànica, a través d'un mecanisme que es veu alterat mitjançant el moviment del mar, com és el cas d'aquest projecte.

La gestió de l'energia adquirida, ha de tenir una alta eficiència per capturar i acumular aquests petits paquets d'energia, alta retenció per emmagatzemar-la durant llargs períodes de temps i la transformació d'aquesta per tal de poder utilitzar-la per a realitzar els treballs desitjats.

Una de les empreses que realitza aquest tipus d'investigació és MELOA, i ho fa mitjançant les WAVY drifters, unes boies llençades al mar que, mitjançant el moviment dels corrents marins i el moviment de les ones té com a finalitat auto alimentar-se mentre va recopilant i transmeten dades in situ.

1.2 PROJECTE MELOA

Les boies de deriva ("drifting buoys") s'utilitzen per a l'estudi de les corrents superficials oceàniques. Són d'utilitat sobretot en zones sense dades oceanogràfiques habituals i de difícil accessibilitat.

Aquest Treball Final de Grau s'emmarca en el context del projecte europeu MELOA (Multi-purpose/Multi-sensor Extra Light Oceanography Apparatus) (https://cordis.europa.eu/project/rcn/212936_en.html) que va començar el desembre de l'any 2017. S'encarrega de desenvolupar una família d'unitats WAVY Drifter, que sigui un sistema d'alta versatilitat, baix manteniment, baix cost i fàcil d'instal·lar, per prendre mesures in situ en entorns marins.

El projecte MELOA proposa desenvolupar una boia de superfície extra lleugera, fàcil de manejar i per tot tipus de medis aquàtics, des d'aigües profundes, àrees costeres fins a zones on és practica surf.

La filosofia WAVY és basa en una esfera de dimensions reduïdes, amb l'espai suficient per allotjar la font d'alimentació, el receptor GPS, els mòduls de comunicació, les antenes, els sensors i el processador de dades. La flotabilitat està optimitzada per evitar que la trajectòria de la deriva respongui al vent en comptes de a la corrent, a l'hora que proporcioni la fiabilitat per assegurar la connexió de la senyal GPS a la velocitat requerida i comunicacions en temps quasi real.

Totes les boies de la família WAVY, han de passar per una sèrie de proves d'acceptació de fàbrica (FAT). Aquestes proves tenen com a objectiu avaluar el rendiment bàsic de tots els seus

components, impermeabilització, duresa superficial, comportament al mar, robustesa, recollida d'energia, fiabilitat de comunicacions, respostes de sensors, processament de dades in situ, etc. Aquestes proves seran realitzades per els socis de MELOA responsables de la fabricació (ISR, OSCAN, CMPSOL, CLS i UPC) juntament amb una col·laboració amb IH i SBI.

En definitiva, MELOA consta d'un objectiu general i 3 objectius específics:

- Objectiu general: innovar en el mercat actual de sistemes de mesura in situ per entorns marins, proporcionant una solució de baix cost per l'adquisició de dades relacionades en forma de família de productes WAVY.
- Objectiu específic 1: Crear una família de productes WAVY que consti de diferents configuracions (diferents sensors, diferents autonomies...) abordant diferents casos d'ús per mesures marines. Dins d'aquest objectiu és pretén que algunes de les configuracions de boia puguin disposar de captadors d'energia ("energy harvester") a partir dels moviments marins.
- Objectiu específic 2: Generar dades valuoses i analitzar-les per programes com GEOSS i Copernicus.
- Objectiu específic 3: Arribar als usuaris potencials dels WAVY drifters per promoure el producte i obrir noves oportunitats per el desenvolupament del mercat a Europa i altres continents.

1.3 MOTIVACIÓ

Un cop hem entès en que és basa el projecte MELOA, quines són les seves propostes, els seus objectius i un cop vist que són els sistemes WAVY, no caldria dir més en quan a motivació a l'hora de realitzar aquest projecte.

Si tenim en compte que la recopilació de dades que és farà a l'energy harvesting (acumulador d'energia el qual parlarem amb més detall en el següent capítol) és la continuació del projecte de final de grau d'enginyeria mecànica dut a terme durant l'any anterior, és una gran motivació extra.

Una de les principals motivacions a l'hora de fer aquest projecte és que la major part és experimental, cosa que per a tot enginyer, o futur enginyer, el poder veure que els esforços i les hores invertides tenen retorn en la rellevància de les dades i els anàlisis obtinguts posteriorment, és un extra de motivació.

A més, un altre aspecte molt rellevant, és el fet d'entendre les necessitats de MELOA, escoltant els requisits en quant a sensors, les dades necessàries i manera d'obtenir-les que haurem de dur a terme al llarg del projecte per poder fer un disseny òptim, amb el menor consum d'energia per la major durada a la deriva.

Referent a les necessitats del sistema de captació de dades a realitzar, no només cal tenir en compte els requisits de MELOA, sinó les proves FAT que haurà de passar la boia. Centrant-se en el nostre cas en la recopilació d'informació.

1.4 OBJECTIUS

L'objectiu principal d'aquest projecte és fer un sistema de recopilació de dades el més robust i senzill possible, amb el menor consum d'energia i la posterior monitorització de les dades obtingudes per una WAVY drifter.

Les dades a recopilar seran per un costat les obtingudes per un acceleròmetre, amb aquest acceleròmetre captarem el moviment de la boia produït per les corrents marines i per les pròpies ones del mar. Un altre dada molt important a recopilar és l'acumulació d'energia que tindrem provinent de l'energy harvesting, aquesta energia la mesurarem mitjançant les pròpies acceleracions mencionades anteriorment i mitjançant uns sensors els quals parlarem mes endavant.

Per poder veure aquesta acumulació d'energia, ho farem mitjançant un sistema de monitorització de dades, explicat i programat al llarg d'aquest projecte amb el programa Labview.

Un aspecte molt important a valorar, és que, degut a que és una boia de deriva, s'haurà de tenir en compte la programació dels sensors i els aparells els quals utilitzarem per tal d'aconseguir el menor consum d'aquests.

Cal mencionar que un dels principals objectius és poder dissenyar, muntar, i programar amb el menor cost possible.

Per tant, l'objectiu principal d'aquest projecte és fer un sistema de captació i monitorització de dades, amb el posterior anàlisi dels resultats obtinguts en un experiment al mar, en una boia de deriva.

1.5 ESTRUCTURA

En primer lloc explicarem el projecte d'enginyeria mecànica de l'any anterior, ja que, és una part vital del nostre projecte. L'energy harvesting dissenyat l'any anterior anirà dins de la nostra boia. En aquest apartat explicarem el disseny, el seu funcionament i les dades que vam adquirir amb el pronòstic que vam fer en el seu moment, per poder comparar si veritablement les conclusions d'aquell moment van ser les adequades.

A continuació explicarem el disseny proposat d'aquest projecte, des del mètode operatiu, fins al tipus de comunicació entre els diferents aparells que utilitzarem, amb una breu explicació de les dades que obtindrem i el tractament que els hi farem per que les dades siguin correctes i coherents.

Tot seguit explicarem els aparells que utilitzarem al llarg del projecte, pin out, metodologies de funcionament, tipus d'alimentació, i els sensors que venen incorporats.

Un cop sabem tot el que necessitem per dur a terme el projecte, explicarem el disseny, des de la programació utilitzada, incloent els diagrames de blocs, i l'explicació de l'aplicació Ubidots, una aplicació per penjar dades in situ a la web.

Posteriorment, un cop sabem tota l'explicació teòrica, passarem a la pràctica, i a la recopilació de dades i proves a realitzar. Veurem els aparells amb els que obtindrem el voltatge que ens proporciona l'energy harvesting i una sèrie de proves que s'han realitzat abans d'anar al mar a fer el test. Per tal de saber quins paràmetres són els adients i quina és la millor configuració per tenir uns bons resultats.

Finalment explicarem la boia construïda en aquest projecte veurem el test al mar, les dades que hem adquirit i unes conclusions finals.

2 ENERGY HARVESTING

Durant el quadrimestre anterior 2018-2019, el projecte de final de grau d'enginyeria mecànica, és basava en el disseny d'un captador energia, aquesta energia l'obteníem mitjançant el moviment del mar per tal de poder alimentar els sensors d'una boia a la deriva, així doncs, aquest projecte de fi de grau d'enginyeria electrònica és la continuació del realitzat l'any anterior.

Tot hi que no ens centrarem gaire en el que és l'energy harvesting (EH), si farem un petit resum, ja que, un dels objectius d'aquest projecte és poder veure l'energia que acumula.

2.1 DISSENY

Com a idea bàsica el funcionament de l'EH dissenyat l'any anterior és la rotació de l'eix d'un generador en un sol sentit, per d'aquesta manera poder generar energia mecànica i transformar-la amb l'ajut d'un generador en energia elèctrica.

El moviment amb el qual és va dur a terme el projecte és mitjançant el moviment pendular. El problema del moviment pendular és que només aprofitem el moviment en una direcció, depenent de l'eix on el posem, podem obtenir moviment en l'eix de les X (roll), moviment en l'eix de les Y (pitch) o moviment en l'eix Z (yaw), tal com mostra en la següent imatge:

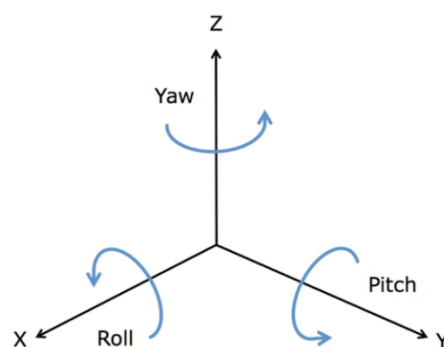


Figura 2.1. Eixos de coordenades i els seus corresponents moments de rotació[Font:www.researchgate.net].

Per aquest motiu, ja que el moviment del mar és completament aleatori i en els tres eixos cartesianes, és va introduir sobre el primer un altre moviment pendular i finalment un tercer sobre aquests dos. El propòsit de tres moviments pendulars és aprofitar el moviment en els tres eixos cartesianes, per que així és vagin alterant entre ells i poder aprofitar tot el moviment produït per l'onatge marí.

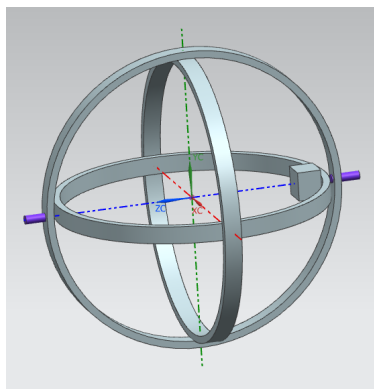


Figura 2.2. Dispositiu amb els dos cercles. El cercle intern amb rotació respecte a l'eix XC i l'extern amb rotació respecte a l'eix YC.

Com es pot veure en la figura anterior el primer pèndul, l'interior, col·locat sobre l'eix de les X (eix que es mostra en la figura de color vermell) és l'encarregat de fer girar l'eix del generador mitjançant una sèrie d'elements els quals mostrarem més endavant.

Sobre aquest primer, trobem un altre arc, que només es veurà afectat amb el moviment de l'eix de les Y (eix de color verd), i finalment com hem explicat anteriorment trobem l'últim arc que es veurà afectat pel moviment en l'eix Z (eix de color blau).

2.2 FUNCIONAMENT

Tot hi que els tres arcs obtindran moviment, és l'arc interior el que fa girar l'eix del generador, per tant l'únic propòsit de l'arc mig i de l'arc exterior, és mitjançant el moviment del mar, pendular per tal de poder alterar el moviment de l'arc interior i d'aquesta manera poder girar l'eix del generador.

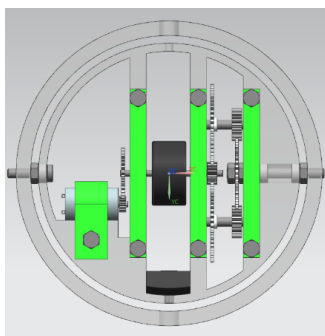


Figura 2.3. Imatge frontal de l'arc interior amb els engranatges.

En la figura 2.3 es mostren els elements que trobem en l'arc interior. Mitjançant una sèrie d'engrenatges, augmentem la velocitat de gir d'entrada respecte a la velocitat de gir que obtenim a la sortida.

Ja que l'objectiu era fer girar l'eix del generador en un sentit, en l'interior de dos d'aquests engranatges trobem un rodament "one way". Es un rodament que només permet el moviment en un sentit, bloquejant-lo en el sentit oposat.

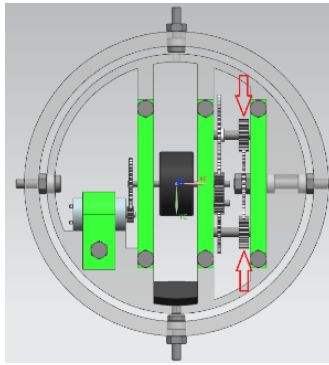


Figura 2.4. Imatge dels rodaments one way.

En l'interior d'aquest arc trobem dos trens d'engranatges, ja que amb el rodament "one way" obtenim moviment només en un sentit, amb aquests dos trens invertits obtenim moviment en els dos sentits de rotació.

Al girar aquest arc en un sentit, trobem que, en un tren d'engranatges es troba el rodament "one way" bloquejat, i en l'altre tren el rodament gira lliurement sobre la barra on es troba fix, és el rodament que es troba bloquejat el que fa girar la barra la qual està fix i transmet moviment a tot el tren d'engranatges. Per altra banda, al girar en el sentit oposat tindrem l'altre tren amb el rodament "one way" girant lliurement, i el segon tren transmetent moviment. Amb la combinació d'aquests dos obtenim moviment en els dos sentits de gir del pèndol, produint el moviment de l'eix del generador en un sol sentit.

En aquesta estructura també trobem una roda d'inèrcia, la funció d'aquesta no és augmentar la velocitat de gir, ja que ve regulada pels trens d'engranatges, sinó poder mantenir la velocitat durant un temps més prolongat.

2.3 DADES ADQUIRIDES

Per finalitzar el projecte es van realitzar una sèrie de simulacions per tal de poder saber quina energia podia arribar a acumular.

Aquestes simulacions es van dur a terme al centre tecnològic de la UPC de Vilanova i la Geltrú. L'assaig consistia en mitjançant una bomba, simular les possibles ones que ens podem trobar al mar, com podem veure en la figura 2.5.



Figura 2.5. Imatges fetes durant la simulació.

Posteriorment amb l'ajuda d'un oscil·loscopi és va treure una gràfica de la potència instantània.

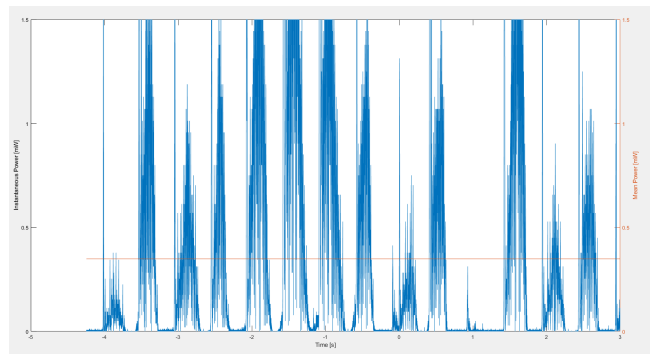


Figura 2.6. Gràfica de la potència instantània.

Els resultats obtinguts és van analitzar mitjançant la funció d'integració d'àrees que proporciona Matlab.

Els resultats van desvelar que al llarg d'un dia l'energy harvesting és capaç d'acumular 30,1071 Joules.

3. PROPOSTA DEL DISSENY ACTUAL

Al llarg d'aquest capítol explicarem com es pretén aconseguir els objectius d'aquest projecte, quin serà el mètode operatiu, els canals de comunicació que utilitzarem per comunicar els diferents aparells de captació de dades, i el tractament que se li aplicarà a les dades obtingudes per tal que els resultats finals siguin els adequats.

3.1 MÈTODE OPERATIU

Per poder veure el moviment produït per la boia un cop és troba sota els efectes dels corrents marins deixarem anar la boia lliure al mar.

Dins de la boia, tindrem un aparell que s'encarregarà de mesurar les acceleracions en tot moment, una placa que utilitzarem com a cervell, per rebre les dades del acceleròmetre, tractar-les i enviar-les a l'ordinador on les mostrarem, l'EH i una bateria de 9V.

El que es vol aconseguir amb l'acceleròmetre és obtenir les acceleracions de la boia, referenciada al seu centre i poder veure el moviment en els tres eixos cartesianes, d'aquesta manera, podrem observar el pitch, el roll i el yaw.

Mitjançant la programació del controlador que utilitzarem com a cervell, decidirem en quin moment enviar, rebre, obtenir el voltatge de la bateria i inclús poder fer un reset al acceleròmetre en cas que ho veiem necessari, podent variar la freqüència i l'escala.

Com podrem veure mes endavant, una de les característiques del controlador és la incorporació d'un wifi, el qual connectarem a un router i podrem enviar la informació a un ordinador, connectat també a la mateixa xarxa.

Aquesta informació l'enviarem mitjançant la comunicació UDP, comunicació molt comú en aquests tipus d'aparells, i que explicarem mes endavant, juntament amb el seu funcionament i la seva programació.

Per tal de mostrar les dades adquirides, mitjançant la connexió wifi i la comunicació UDP a l'ordinador, utilitzarem el programa Labview.

Amb aquest programa mostrarem per pantalla de manera iterativa i molt senzilla el moviment de la boia.

Per acabar el que es pretén és pujar les dades adquirides a una pàgina web. Per tant, amb el programa LabView, veurem les dades d'una manera ordenada i molt senzilla, mentre que per la pàgina web, obtindrem els resultats de les acceleracions de la boia.

3.2 CANALS DE COMUNICACIÓ

3.2.1 COMUNICACIÓ SÈRIE

Aquesta comunicació és la que utilitzarem entre l'acceleròmetre i el controlador.

La comunicació sèrie utilitza un sistema de transmissió asíncrona, es a dir, les dades s'envien sense que el remitent enviï una senyal de rellotge, d'aquesta manera el receptor i el emissor han de estar configurats amb la mateixa velocitat de transmissió, característica vital per el funcionament del port sèrie asíncron, de manera contraria les dades rebudes son il·legibles.

Per assegurar que la transmissió de dades segueix aquest sistema, tant l'emissor com el receptor utilitzen els seus propis circuits interns, agregant bits especials en cada una de les dades enviades.

El bit d'inici, s'introdueix al començament de cada transmissió, amb el propòsit d'alertar al receptor que la transmissió és apunt de començar, a mes, obliga al rellotge del receptor a sincronitzar-se amb el rellotge del transmissor.

Per poder enviar dades necessitem un hardware que estigui dissenyat per dur a terme aquesta tasca, i en el nostre cas s'anomena UART (Universal Asynchronous Receiver/Transmitter). UART és el dispositiu electrònic destinat a la transmissió de dades de manera asincronia i en sèrie.

Tot i que el seguit d'instruccions que veurem a continuació ho fa el propi chip integrat mitjançant les seves llibreries, veurem un exemple per tal de comprendre mes fàcilment en que consisteix l'enviament i recepció de dades.

Imaginem que volem enviar el caràcter "a" per el port sèrie, el primer que es fa és buscar quin és el número associat al codi ASCII (American Standard Code for Information Interchange), en el nostre cas trobarem que el valor correspon a 97.

Si convertim el 97 a binari, obtenim que la lletra "a", el número 97 en ASCII, equival a 01100001.

Això és el que realment tenim que enviar pel port sèrie, aquesta combinació d'uns i zeros, però d'una manera especial, dins d'una trama.

El primer que es fa per enviar les dades és invertir l'ordre del codi binari, és a dir, el primer passa a ser l'últim. Això és degut que en aquest protocol primer s'envia el bit menys significatiu.

La línia de transmissió sempre que es troba en repòs està en nivell alt.

El primer bit que s'envia es un bit especial, l'anomenat bit d'inici, que sempre serà un 0 mantenint-lo durant un període de temps, anomenat temps de bit.

Un cop passat el temps de bit enviarem el codi binari mostrat anteriorment, com es pot veure en la figura 3.3.

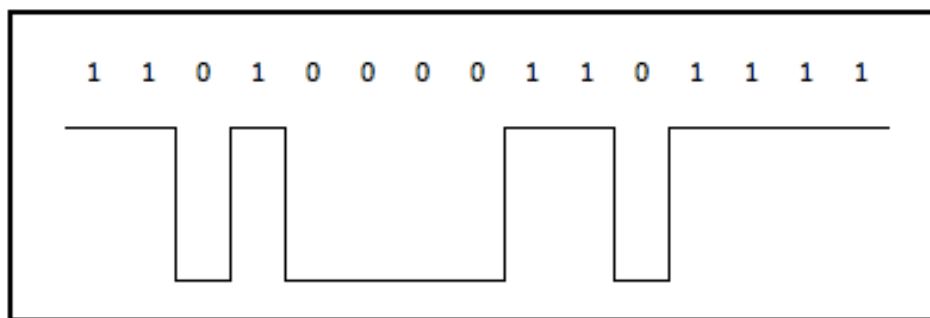


Figura 3.1. Codi binari a enviar "97".

Cada bit estarà en la línia de dades durant el temps de bit el qual hem parlat anteriorment.

Per indicar que la trama ha acabat, UART envia l'anomenat bit final, el qual és de 0 a 1 durant un mínim de dos bits de duració, i amb aquest fet, la línia romandrà en nivell alt esperant el següent bit d'inici.

El temps de bit anomenat anteriorment és el temps que mantindrem un bit en la línia de transmissió, el seu càlcul és molt senzill i està relacionat directament amb els Baudis (numero de bits per segon que es poden enviar).

El temps de bit correspon a la inversa dels baudis, per una comunicació de 9600 baudis, el temps de bit correspon a:

$$Tb = \frac{1}{\text{Baudis}}$$

$$Tb = \frac{1}{9600} = 104,2 \mu S$$

Per tant cada un dels bits enviats és mantindrà 104,2 μS en la línia.

En el nostre paquet de dades, tenim 10 bits, un bit d'inici, 8 bits de trama i un bit de final de transmissió.

Per acabar amb la comunicació sèrie, només ens cal veure com s'envien aquests conjunt de 0 i 1, aquesta enviament és fa mitjançant impulsos de voltatge.

La trama del port sèrie pot ser enviada mitjançant molts estàndards, però la utilitzada per els microcontroladors es l'anomenada com a nivells TTL.

Els nivells TTL significa que el 0 i els 1 els obtindrem com a nivells de voltatge, representats a 0 volts per els 0 i en el nostre cas a 3,3 V per els 1.

3.2.2 COMUNICACIÓ UDP

La comunicació UDP ha estat l'escollida per la transmissió de dades entre el controlador i l'ordinador, en el qual analitzarem les dades rebudes mitjançant el programa Labview.

El protocol de datagrames d'usuari (UDP) proporciona un servei sense connexió, amb un flux de dades molt ràpid, en canvi, per contra no dona garantia a l'entrega ni comprova la seqüència dels datagrames.

Per poder utilitzar la comunicació UDP necessitem els ports UDP. Un port UDP proporciona una ubicació per enviar i rebre missatges. Funciona com una única cua de missatges que rep tots els datagrames destinats al programa especificat, mitjançant cada número de port del protocol.

El valor del port el pot escollir l'usuari tenint en compte que té una llargada màxima de 16 bits. El rang va des de 0 a 65535, i a més hi ha alguns ports que es troben reservats.

- Els ports que es troben entre el 1 i el 1023 són els coneguts com a ports "bien conocidos" i en alguns sistemes operatius enllaçar amb aquests ports requereix el permís de superusuari.
- Els ports compresos entre el 1024 i el 49151 són ports registrats.
- Per acabar els que es troben entre el 49152 i el 65535 són ports efímers i es solen utilitzar com a ports temporals sobretot per els clients al comunicar-se amb els servidors.

Per tant els ports escollits per transmetre les dades estan compresos entre el 49152 i el 65535.

El format del missatge UDP el podem veure en la següent imatge:

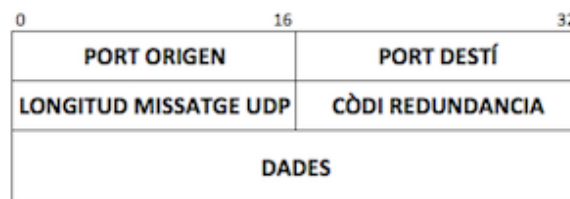


Figura 3.2. Codi binari a enviar "97". [Font: Wikipedia].

- Port UDP origen: 16 bits, opcional. Número del port del microcontrolador que envia el missatge.
- Port UDP destí: Port destí 16 bits. Número de port de la màquina a la qual va destinat el missatge.
- Longitud del missatge: 16 bits. Especifica la longitud mesurada en bytes del missatge UDP.
- Codi redundància: Es una comprovació bastant dèbil amb una alta probabilitat a error, per tant, si volguéssim una connexió molt segura, aquest mètode, com hem dit anteriorment, no seria l'adequat.
- Dades: En aquest camp es troben les dades que enviem des del node al ordinador per que Labview les capti i posteriorment les analitzi.

La connexió UDP és una connexió complexa de programar, per aquest motiu, controlador compta amb una llibreria la qual, ens facilita molt la feina a l'hora d'utilitzar aquesta comunicació.

La llibreria WifiUDP, conté funcions tals com:

- `WifiUDP.begin`: Inicialitza la llibreria i configura la connexió de la xarxa.
- `Available`: obté el número de bytes disponibles per la lectura del buffer. Es tracta de dades que ja han arribat.
- `WifiUDP.beginpacket`: inicia la connexió per escriure dades UDP.
- `WifiUDP.write`: aquesta funció escriu dades UDP en la connexió remota.

Totes aquestes i moltes altres funcions ens faciliten molt la feina a l'hora de programar, rebre i enviar dades mitjançant la comunicació UDP.

Per que hem escollit UDP i no TCP?

En general, la diferencia és com entreguen les dades, l'exemple perfecte, seria entre el d'una trucada telefònica i una postal. TCP funciona com una trucada telefònica, comprova que el destí estigui disponible i preparat per la comunicació. D'altra banda UDP funciona com una postal, els missatges son petits i l'entrega probable, però no sempre està amb garantia que arribi.

- UDP no estableix una connexió entre hosts, mentre que TCP sí.
- UDP no garanteix l'entrega, mentre que TCP sí.

En aquest cas doncs perquè UDP i no TCP?

Precisament per l'última diferència, UDP és molt més ràpid, envia paquets a una velocitat molt superior que no la comunicació TCP i a l'hora té molts menys requisits de carga.

3.2.3 UBIDOTS

Ubidots va néixer com una empresa privada de serveis d'enginyeria en l'any 2012, es va especialitzar en solucions connectades en hardware i software per monitoritzar, controlar i automatitzar de manera remota els processos, per els clients d'atenció medica en empreses del sud est d'estats units i tota Amèrica llatina.

Entre el 2012 i el 2014, va participar en innumerables projectes connectats a internet relacionats en salut, energia, serveis públics, fabricació, transport, etc.

En l'any 2014 Ubidots es va donar a conèixer en hardware, software i en enginyeria integrada com la plataforma més assequible, confiable i més utilitzada en l'entorn dels dispositius IoT.

Finalment en l'any 2018, Ubidots va desplegar la plataforma "for education", per blindar als estudiants de IoT un lloc on construir, desenvolupar, provar, aprendre i explorar el futur de les aplicacions i solucions connectades a internet.

Ubidots per tant, com veurem mes endavant, ens permet amb una sèrie de comandes i llibreries penjar les dades a la Web, per poder analitzar-les i veure-les en tot moment, podent-nos connectar des de l'ordinador o des de qualsevol dispositiu amb connexió a internet, ja sigui tableta, telèfon mòbil, etc.

3.3 TRACTAMENT DE DADES

3.3.1 ACCELERACIÓ

Un cop vist com seran els protocols de comunicació i on i com veurem les dades, cal saber quines dades adquirirem i quins tractaments farem per tal que les dades rebudes siguin reals, i obtinguem els resultats adients.

Els valors obtinguts per l'acceleròmetre, son valors d'acceleració. Aquests valors en els tres eixos, seran dividits per pel valor màxim captat pel sensor. D'aquesta manera obtindrem el valor de l'acceleració en forces G, que es converteixen en unitats del sistema internacional multiplicant-los per el valor de la força de la gravetat ($9,80665\text{m/s}^2$).

$$\text{Acceleració en G} = \frac{1}{\text{Valor màxim}} * \text{Valor llegit del sensor}$$

$$\text{Acceleració en } m/s^2 = \text{Acceleració en G} * 9,80665$$

Per saber el valor dels angles mitjançant l'acceleració s'ha de recórrer a la física.

En el 2D, imaginem que posem el sensor horitzontal al pla X-Z, apuntant cap amunt, i únicament girem entorn l'eix Y, l'equació resultant és:

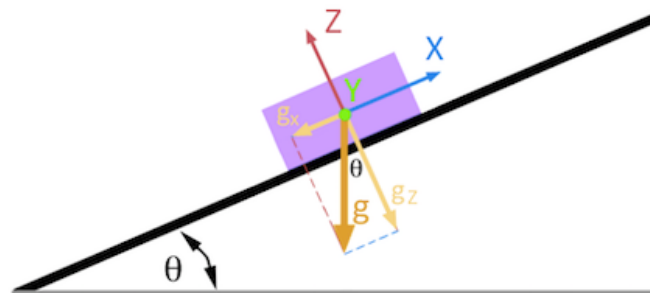


Figura 3.3. Acceleracions 2D. [Font: www.luisllamas.es].

$$\theta = \text{atan}\left(\frac{g_x}{g_z}\right)$$

I així per cada eix. En el cas del 3D la formula és una mica diferent.

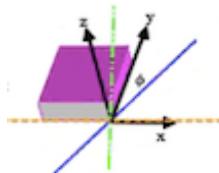


Figura 3.4. Acceleracions 3D. [Font: www.luisllamas.es].

Sabent que, es produirà una força contrària degut a l'efecte de la gravetat a les fletxes que veiem en la imatge anterior, i que la suma de les acceleracions $A_x + A_y + A_z = g$, obtindrem el següent sumatori de vectors:

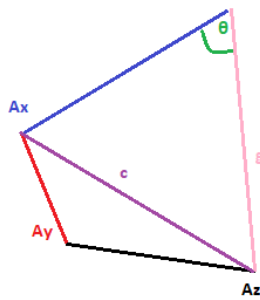


Figura 3.5. Trigonometria acceleracions 3D.

Utilitzant trigonometria sabem que:

$$\varnothing = \frac{Ca}{Co} = \frac{Ax}{c}$$

Mitjançant l'algoritme pitagòric sabem que el valor de c equivalen a l'arrel quadrada de la suma dels catets.

$$c = \sqrt{Ay^2 + Az^2}$$

Per tant, per saber l'angle en tot moment haurem de fer la següent operació per cada instant de temps:

$$\varnothing_x = \frac{Ax}{\sqrt{Ay^2 + Az^2}}$$

I el mateix per l'eix Y:

$$\varnothing_y = \frac{Ay}{\sqrt{Ax^2 + Az^2}}$$

Un problema que tenim a l'hora de passar les acceleracions a angles és que mai podrem veure l'angle en l'eix Z, això es degut a que calculem l'angle mitjançant l'acceleració, que en aquest cas és la força de la gravetat. Si movem el sensor en torn a l'eix Z, la força de la gravetat sempre serà paral·lela a l'eix, per tant, l'angle que ens donarà en aquest eix serà sempre nul.

3.3.2 TRANSFORMADA DE FOURIER

Per entendre la Transformada de Fourier, primer hem d'entendre que es una ona simple.

Una ona simple es basa en 3 conceptes:

- Amplitud: altura de la ona.
- Freqüència: número d'ondulacions per segon.
- Desfase: moment en que comença l'ona.

Per tant, una ona simple podria ser com la de la imatge següent:

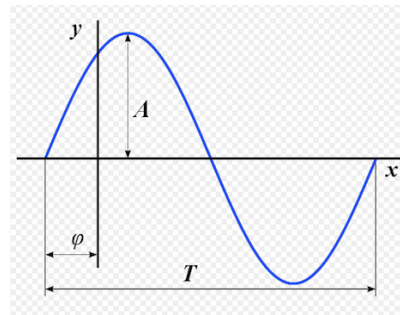


Figura 3.6. Ona simple amb desfase. [Font: Wikipedia].

En la següent imatge, podem veure un conjunt d'ones simples.

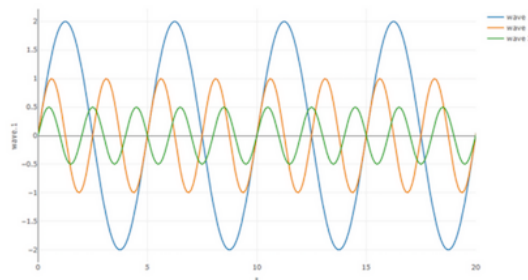


Figura 3.7. Conjunt d'ones simples. [Font: www.conceptosclaros.com].

Aquest conjunt d'ones simples, com podem veure, tenen una cosa en comú, el desfase es 0, d'altra banda les dos següents caracteritzacions d'ones simples les tenen diferents, és a dir, tenen diferent amplitud (cap té la mateixa altura) i tenen diferent freqüència (al llarg d'un segon, cap ona es repeteix el mateix nombre de vegades).

Una sèrie temporal, es un conjunt d'observacions d'una variable ordenada cronològicament, és a dir, un conjunt d'ones simples.

Doncs be, el que va descobrir Fourier, és que es pot separar una ona temporal, com un conjunt d'ones simples.

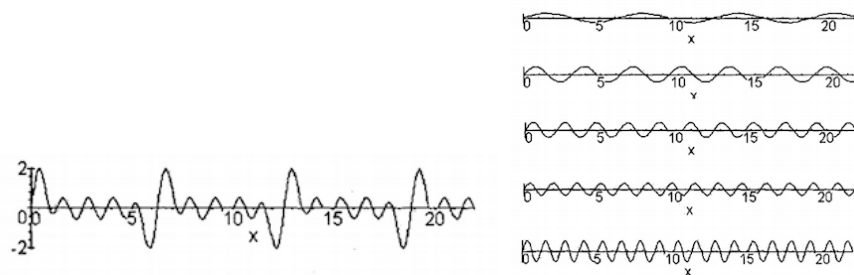


Figura 3.8. Sèrie temporal i conjunt d'ones.

Com podem veure en la imatge 3.10 (Esquerra) es el conjunt de les ones simples de la dreta, o dit d'una altra manera, el sumatori de les ones de la imatge 3.10 (dreta) es igual a l'ona de la imatge de l'esquerra.

La finalitat de separar les ones temporals en ones simples, es que ens permet dibuixar una ona temporal en funció de la freqüència de les ones simples, d'aquesta manera, és més fàcil visualitzar la magnitud de la ona (amplitud) i el ràpid que van (freqüència).

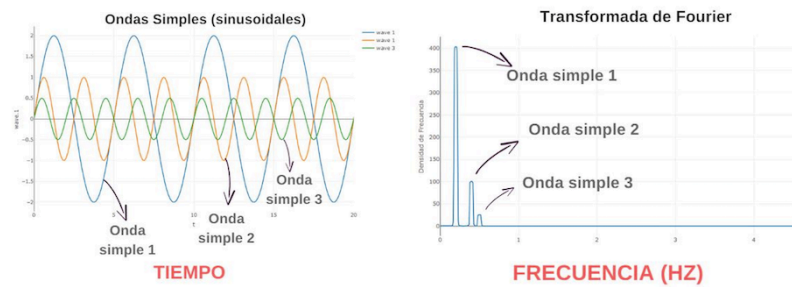


Figura 3.9.Transformada de Fourier. [Font: www.conceptosclaros.com].

Per tant, amb la transformada de Fourier, obtenim el mapa de freqüències, on queda condensada tota la informació que has obtingut de les ones simples provinents de l'ona temporal.

En resum, podem passar d'una sèrie temporal complicada, a una suma d'ones, i podem utilitzar el mapa de freqüències utilitzant la freqüència, com a mesura per diferenciar les diferents ones.

Cal entendre bé el que és i el que s'aconsegueix amb la transformada de Fourier, ja que, més endavant, al llarg de l'anàlisi dels resultats, farem la transformada de Fourier per obtenir el mapa de freqüències que obtindrem d'una ona temporal provinent de l'energy harvesting.

4 REQUISITS DEL DISSENY

4.1 WAVY DRIFTERS

Les dimensions que tindrà la boia oscil·len entre els 200 i 250 mm de diàmetre.

Es tractarà d'una boia amb un pes de 3,4 kg, un pes molt elevat per una boia. Això es degut a que es vol que la boia vagi molt enfonsada, amb la finalitat que no quedi gaire part destapada per sobre del nivell del mar. L'objectiu de tot això és que la boia no sigui influenciada pel vent, que tingui un efecte d'abatiment el més baix possible i que es desplaci amb els corrents marins.



Figura 4.1. Boia projecte MELOA. [Font: MELOA].

Al tractar-se d'una boia a la deriva, aquesta serà arrossegada per les ones i corrents marines superficials. El seu moviment serà la suma de les forces que actuïn sobre ella.

D'altra banda, els moviments que es donen a l'oceà tenen una gran varietat d'escala tant temporals com espacials, des de fenòmens de centímetres i segons, fins a processos de circulació global entre conques oceàniques.

Les forces que actuen sobre un element fluid de l'oceà són fonamentalment: la força de la gravetat, la força de Coriolis, les forces de fricció i les forces degudes a variacions de pressió.

Es el vent com procés atmosfèric, una de les principals raons del moviment en la superfície oceànica. El vent incideix sobre una prima capa de la superfície oceànica generant corrents marines superficials.

Segons Cardona Díaz (2015), el desplaçament d'una boia de deriva es pot representar principalment com la suma de l'efecte de la corrent geostrofica i del vent, a més d'altres fenòmens que no es poden mesurar directament. Així, la velocitat de qualsevol objecte a la deriva (V_d) es pot escriure com:

$$V_d = aW + bC + \delta$$

On W i C són els vectors de vent i de corrent respectivament. Mentre que a i b són coeficients de ponderació que depenen de la relació de les àrees exposades a l'acció del vent i de la corrent. Essent delta un terme d'incertesa aleatori.

Els valors dels coeficients a i b poden canviar en varis ordres de magnitud en funció de l'objecte a la deriva, depenent de quina part de l'objecte queda submergida i per tant exposada a la corrent i quina part queda exposada al vent.

4.2 TD1205P

Aquesta es la placa escollida amb la que obtindrem les acceleracions.

Aquesta placa es caracteritza pel seu baix consum i per la gran quantitat de sensors que té incorporats, entre altres, podem obtenir, la latitud, la longitud, la temperatura i com hem dit, les acceleracions.



Figura 4.2.TD1205P. [Font: Datasheet TD1205P].

La programació d'aquest dispositiu l'ha realitzat SARTI, amb una sèrie de comandes i opcions les quals veurem mes endavant per poder variar la freqüència i l'escala, per tant, al llarg d'aquest projecte no centrarem l'atenció en com s'ha programat d'aquest dispositiu, però sí en com funciona, i com rebrem la informació.

4.2.1 PINOUT

El pinout del TD1205P consta de 15 pins, disposats al llarg de la placa com podem veure en la següent imatge:

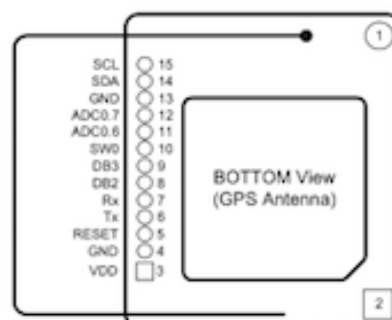


Figura 4.3.Pinout TD1205P. [Font: Datasheet TD1205P].

Els pins 1(-) i 2 (+) son pins utilitzats per alimentar el TD1205P amb una font d'alimentació externa, aquest dispositiu igual que el ESP8266, permet una entrada màxima de 3.6V, essent el voltatge de funcionament de 3.3V, la mateixa funcionalitat tenen els dos pins següents, el 3 per l'alimentació i el 4 pel GND.

El pin 5 és un pin el qual utilitzarem al llarg d'aquest projecte per poder fer un reset des de l'entorn Labview, en el moment en que ho desitgem. Aquest pin conté una resistència pull-up

interna i sempre és manté en estat alt, d'aquesta manera si no l'utilitzéssim el podríem deixar a l'aire.

Els pins 6 i 7 son els encarregats de la comunicació sèrie, com hem explicat anteriorment, son els que aniran connectats (RX-TX i TX-RX) amb el node ESP per la transmissió de dades.

Els tres pins següents, 8 DB2, 9 DB3 i 10 SWO, son pins els quals no entrarem en detall, ja que com hem dit anteriorment al llarg d'aquest projecte no és realitza la programació del TD1205P. Per saber una mica mes d'aquests pins, son uns pins utilitzats per la depuració, el pin DB2 és l'entrada i sortida de dades, s'habilita després de reiniciar-se i conté un pull-up incorporat. El pin DB3 és l'entrada de rellotge amb cable sèrie, aquest pin s'habilita també després de reiniciar i a mes, conté un menú desplegable per la seva programació. Per acabar, el pin SWO serveix per fer una espècie de seguiment completament independent de la CPU. Aquets tres pins es poden configurar per enviar comandes AT, les quals veurem en el següent apartat, tot i que al llarg d'aquest projecte aquestes comandes les enviarem per port sèrie, mitjançant els pins TX i RX del node ESP.

Per acabar ens trobem amb un altre pin de terra i els pins SDA i SCL, els quals no s'utilitzaran. Aquets son per la programació i transferència de dades mitjançant I2C, una configuració mestre-esclau.

Tots els pins TX, RX, DB2, DB3, SWO, SDA i SCL poden configurar-se com a triple estat push-pull, drenatge obert i amb o sense resistència pull-up, a més, és pot configurar amb una potencia d'accionament programable de 0,5 mA/ 2 mA/ 6 mA/ 20 mA.

Si és configuren com entrades, en aquests pins ve incorporat un filtre de supressió de fallades opcional.

4.2.2 ACCELERÒMETRE

Com hem explicat anteriorment, el TD1205P té 3 sensors, GNSS, acceleròmetre i un sensor d'efecte hall.

Si veiem el datasheet d'aquest chip, podem veure que l'acceleròmetre té un rang de mesura de +- 2g fins a +- 16g, una sensibilitat que és mou entre 1 i 12 mg/dígit, depenent de la configuració, i una sensibilitat vers la temperatura de 0,01%/ °C , essent la temperatura de funcionament idònia de 25°C .

4.2.3 COMANDES ATT PROGRAMADES

La programació del TD1205P, ens permet fer una sèrie de variacions programades mitjançant les comandes AT.

La velocitat de transmissió per defecte és de 460800 baudis, tot i que pot variar en els rangs de 9600 fins a 460800.

L'escala per defecte és de 2G i la freqüència de 10 HZ.

Per poder canviar la freqüència, necessitem enviar la comanda: AT\$ACCFREQ=

- 1 a 1 Hz.
- 2 a 10 Hz.
- 3 a 25 Hz.
- 4 a 50 Hz.
- 5 a 100 Hz.
- 6 a 200 Hz.
- 7 a 400 Hz.
- 8 a 1,25 KHz.

Per canviar l'escala, utilitzarem la comanda: AT\$ACCSCALE=

- 1 per 2G.
- 2 per 4G.
- 3 per 8G.
- 4 per 16G.

Per acabar tenim la funció d'habilitar o des habilitar l'acceleròmetre, mitjançant la comanda AT\$ACCDATA=

- 0 per des habilitar l'acceleròmetre.
- 1 per habilitar-lo.
- 2 habilita o des habilita depenent de l'estat en que estigui en el moment.

Per tant, com veurem en el capítol de programació, una comanda per poder rebre dades al nostre controlador és AT\$ACCDATA=1.

4.2.4 COMUNICACIÓ SIGFOX

El node TD1205P està capacitat per utilitzar la comunicació sigfox.

Tot i que en aquest projecte no utilitzarem aquesta cobertura, és una alternativa a la red wifi.

Sigfox és una companyia francesa fundada l'any 2009 que proporciona un servei de xarxa de cobertura molt amplia per aparells de baix consum amb una taxa de transferència fins a 12 bytes.

Aquesta xarxa, funciona amb una tecnologia de transmissió UNB, amb un funcionament molt similar al de la telefonia mòbil, ja que a que aquesta xarxa funciona a partir de la col·locació de varies estacions receptores i transmissores, amb la diferència que la xarxa Sigfox no és troba només a una sola estació base específica, és a dir, qualsevol estació pot rebre informació i transferir-la cap al cloud.

4.3 NODE MCU ESP8266

En primer lloc, cal entendre la diferència entre el Node MCU ESP8862 i el ESP8862, el Node MCU recull tant el firmware (lògica primària que exerceix el control dels circuits) com una placa de desenvolupament basats en el ESP8862. Amb aquesta placa de desenvolupament, obtenim una sèrie d'avantatges els quals no tenim amb el simple chip ESP8862, aquests avantatges són els següents:

- Port micro USB i convertidor sèrie-USB.
- Programació senzilla a través del Micro-USB.

- Possible alimentació a través del USB.
- Pins als terminals per facilitar la connexió.
- Led i boto de reset integrats.

Hi ha tres principals fabricants del Node MCU, Amica, Lion/Wemos i DOIT/SmartArduino. Aquestes plaques son molt similars, tot i que poden tenir alguna diferència en la designació dels pins.

Al llarg del temps, hem pogut veure 3 versions d'aquesta placa la qual veurem a continuació.

4.3.1 GENERACIONS DEL NODE MCU

La primera versió del Node MCU, és va denominar devokit v0.9, i estava integrada sobre un ESP12 amb una memòria flash de 4MB (la memòria del ESP8266 és externa i s'ha de connectar per SPI). El ESP12 és molt similar al ESP12E, l'única diferència és que té una filera menys de pins, d'aquesta manera tenim una fila menys de GPIO.

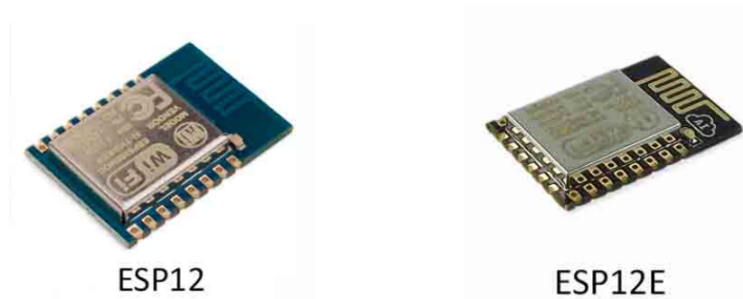


Figura 4.4. Generacions ESP. [Font: Luis Llamas].

La versió 0.9 tenia unes dimensions de 47x21 mm i ocupava 10 files de pins d'una placa breadboard, això la feia molt poc pràctica, ja que no deixava pins lliures per realitzar connexions.



Figura 4.5. Generacions ESP. [Font: Luis Llamas].

La segona generació és va denominar V1.0 /V2, de manera resumida, Amica, una companyia creada per l'Alemanys Gerwin Janssen, va fabricar la seva pròpia versió millorada de la v0.9. A l'equip de Node MCU li va agradar molt i la van declarar com a versió oficial de NodeMCU.



Figura 4.6. Generacions ESP12E. [Font: Luis Llamas].

La principal diferència és que aquesta té un ESP12E en comptes del ESP12, d'aquesta manera deixa més pins disponibles que el model original (v0.9).

A més, és més estreta que la versió anterior, tapant només 8 files d'una breadboard. Això deixa una filera addicional a cada costat per realitzar connexions.

Per acabar, arribem a l'última versió, la tercera generació, 1.0 V3. Bàsicament, el fabricant Lolin/Wemos va decidir crear el seu propi disseny millorat amb uns petits canvis sobre l'anterior.



Figura 4.7. Última Generació ESP. [Font: Luis Llamas].

El principal canvi que ens trobem en aquesta versió és que té un convertidor serial CH340G, en comptes del CP2102, en el que el port USB és més robust. D'altra banda, van reutilitzar els dos pins reservats en la V2 per poder treure un GND i un VUSB.

El pin out d'aquesta última versió és el següent:

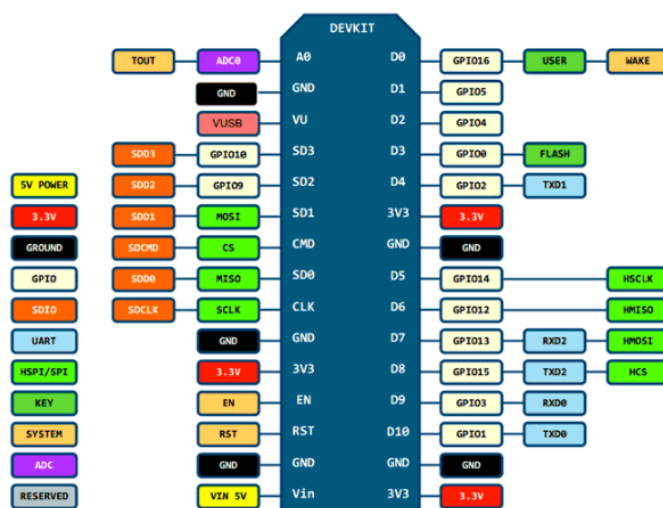


Figura 4.8. Pinout ESP. [Font: Datasheet ESP].

Els pins els podem veure en la imatge 4.8.

- El GPIO15 sempre es troba en pulled down.
- El GPIO0, sempre es troba en HIGH durant el seu funcionament.
- El GPIO2 no pot estar posat a LOW durant l'arrencada.

El ESP8266 té 17 pins I/O (GPIO, general purpose Input/Output pins). Com hem dit anteriorment, poden subministrar tensions de 0V a 3.3V (LOW i HIGH).

Com entrada, els pins no son tolerants a 5 V. Hi ha molts pins els quals no podem utilitzar, de fet, podem utilitzar bastants pocs, un desavantatge respecte a arduino.

- Els GPIO del 6 al 11 s'utilitzen per connectar per SPI amb la memòria flash, així que no els podem utilitzar.
- Els GPIO 0, 2 i 15 intervenen en l'arrencada del ESP, amb les característiques que hem vist anteriorment.
- EL GPIO1 i GPIO3 son utilitzats per la comunicació Serial (UART).

En quant a resistències internes Pull, els pins GPIO0 fins el GPIO15, tenen resistències Pull-up, mentre que el GPIO 16 té una resistència Pull-Down.

La freqüència per defecte és 1kHz, tot hi que la podem variar.

El preu d'aquestes plaques de desenvolupament és molt econòmic, perfectes per projectes de baix cost, oscil·la entre els 2-2,5 euros, la placa V3 sol ser una mica més barata, segurament per muntar sobre un CH340G o per una producció en sèrie major.

Si es vol una placa per muntar un projecte sense breadboard, l'opció preferida sol ser la V3, d'altre banda si el projecte inclou una breadboard, es sol escollir la V2 ja que amb l'amplada inferior, solen ser mes fàcils les connexions.

Degut a que al llarg d'aquest projecte utilitzarem la tercera generació, la de Lolin, entrarem una mica més en detall amb aquest mòdul.

4.3.2 ESP-12E MÒDUL

La placa de desenvolupament ve equipada amb el mòdul ESP-12E marcat de color blau en la següent imatge i conté el chip ESP8266 amb un microprocessador RISC LX106 de 32 bits Tensilica Xtensa® i una freqüència de rellotge ajustable de 80 a 160 MHz.



Figura 4.9. Generacions ESP. [Font: yeoman.com].

Equipat amb 128 KB de memòria RAM i 4 MB de memòria Flash (per l'emmagatzemen de programes i dades).

El ESP8266 integra un transceptor Wifi HT40 802.11b/g/n, amb el que no només pot connectar-se a una xarxa Wifi e interactuar amb internet, sinó que també pot configurar una xarxa pròpia, permetent que altres dispositius es connectin directament a ella.

4.3.3 ALIMENTACIÓ

El rang de voltatge del ESP8266 és de 3V a 3.6V, la placa ve amb un regulador de voltatge LDO (quadre blau de la imatge 4.11) per mantenir el voltatge constant a 3.3.

Pot subministrar fins a 600 mA.

La sortida del regulador és divideix en ambdós costats de la placa (quadres vermells de la imatge 4.11) i permet subministrar energia als components externs connectats a ella, com per exemple el TD1205P en el nostre cas.

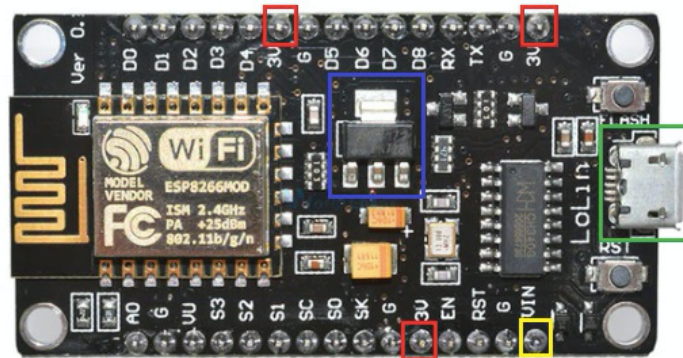


Figura 4.10. Generacions ESP. [Font: yeoman.com].

L'alimentació es subministra a través del connector Micro USB incorporat (quadre verd de la imatge 4.11). Alternativament, si tenim una font de voltatge regulable de 20V, el pin Vin (quadre groc de la imatge 4.11) es pot utilitzar per alimentar directament el ESP8266 i els seus perifèrics a 20V, tot i que nosaltres l'alimentarem a 9V.

El que és molt important i hem de tenir en compte és que els pins GPIO del nostre microcontrolador no son tolerants a 5V.

4.3.4 ARRENCADA

A l'hora de l'arrencada, el ESP8266 té 3 mètodes:

- UART Bootloader, per pujar un programa per UART a la memòria flash.
- Boot Sketch, executa l'últim programa pujat a la memòria flash.
- SDIO, no es pot utilitzar amb el programador d'arduino.

Mètode	GPIO15	GPIO0	GPIO2
Uart Bootloader	0V	0V	3,3V
Boot Sketch	0V	3,3V	3,3V
SDIO	3,3V	x	x

Figura 4.11. Tipus d'arrencada ESP.

En la majoria de plaques de desenvolupament, com és el nostre cas, hi ha sistemes que gestionen automàticament l'estat d'aquests pins. No obstant, si el volem utilitzar de forma independent, tindrem que tenir en compte el pinout del següent apartat.

4.3.5 METODOLOGIES DE FUNCIONAMENT DE BAIX CONSUM

El CHIP ESP8266 consta de 3 configuracions de “sleep modes”, aquestes ens permeten un consum molt baix mentre no utilitzem el node, per tal que no consumeixi bateria, i tingui una autonomia més elevada, aquestes 3 configuracions son:

- Modem-sleep
- Light sleep
- Deep-sleep

Ítem	Modem-sleep	Light-sleep	Deep-sleep
Wi-Fi	OFF	OFF	OFF
System clock	ON	OFF	OFF
RTC	ON	ON	ON
CPU	ON	Depen Configuració	OFF
Consum	15 mA	0,9 mA	20 µA

Figura 4.12. Metodologia Baix consum.

El mode Deep-Sleep és la configuració que proporciona menys consum de bateria, per entrar en aquesta configuració, hem de sacrificar la connexió Wifi, el rellotge del sistema i la unitat central de processos o CPU.

Únicament tindrem habilitat el funcionament del rellotge en temps real (Real Time Clock o RTC). Com podem veure en les 3 configuracions aquest rellotge es troba habilitat, aquest, serà qui envii la senyal per que es pugui reiniciar el processador, quan finalitzi el compte enrere del temps de suspensió que s’ha programat.

Cal comentar que el temps màxim que pot estar en qualsevol de les tres configuracions és de 71 minuts i 30 segons.

Això és degut a que els microprocessadors ESP8266 treballen a 32 bits, per tant el número més alt que és pot introduir es $2^{32} = 4.294.967.295 \mu s$.

4.3.6 SENSOR INFRAROIG

Per poder veure a quina velocitat gira l’eix del generador, hem utilitzat uns sensors d’infraroig.

Els sensors d’infraroig, son uns components elèctrics basats en un led infraroig i un fotoresistor, col·locats un en front de l’altre, de manera que el led actua com emissor i el fotoresistor com a receptor.

El led infraroig emet una llum de major longitud d’ona, (o menor freqüència) que la que els éssers humans som capaços de captar, d’aquesta manera, aquesta llum es invisible per nosaltres.

Si aquesta llum impacta contra una superfície blanca, es reflectirà i arribarà al fototransistor.

Si aquesta llum per el contrari, impacta contra una superfície negra, el material absorbirà la majoria de llum i no arribarà al fotoreceptor.

Existeixen dos tipus de configuració segons el propòsit i la informació que se li vulgui enviar al microcontrolador.

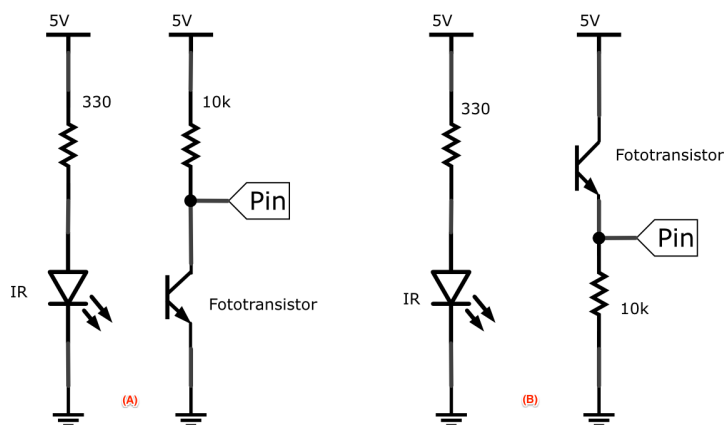


Figura 4.13. Configuració sensors infraroig [https://hardwarehackingmx.files.wordpress.com].

En l'esquema que podem veure en la imatge anterior, podem veure els dos tipus de configuració, bàsicament consisteix en mantenir el led infraroig encès i estar monitoritzant a través del fototransistor, que com el seu nom indica la seva funció és similar a la d'un comú, treballa com a comporta tancada al estar en repòs i oberta quan és excitat per el raig de llum infraroig.

En el nostre cas utilitzarem la opció B, aquesta configuració es un clàssic pull down, en que la sortida del pin proporciona un estat baix (LOW) quan el fototransistor es troba en repòs (tancat), i un estat alt (HIGH) quan està excitat (obert), lògicament la opció A funciona de manera inversa.

4.3.7 ADP5092 MAXIMUM POWER POINT TRACKING

Hem de tenir en compte que la potència que obtindrem en el motor, dependrà de la càrrega que hi hagi en els borns, dit d'una altra manera, dependrà del corrent que se li extregui, és a dir, dependrà del voltatge que tingui en borns. Per tant, podem dir que, el fet que motor giri a certes revolucions, no vol dir que tingui una potència fixa, ja que dependrà de la càrrega que tingui en aquell moment.

Per aquest motiu, utilitzarem un power management unit. El que fem amb aquest aparell electrònic, el qual podem veure en l'article de Matias Carandell adjunt a la biografia, és fixar el voltatge en borns del motor, i així, obtenir la màxima potència.

Aquest seguiment se li diu maximum power point tracking, la PMU (Power Management Unit) utilitzada és la ADP5092.

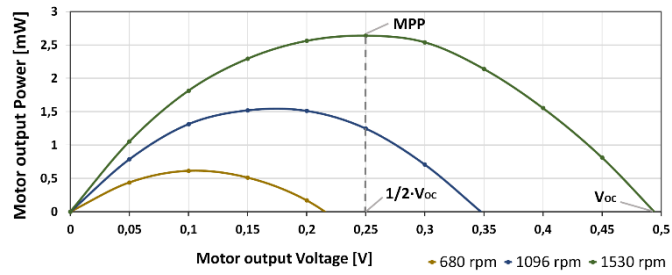


Figura 4.14. Curves del Generador. [Font: Matias Carandell].

El que fa aquesta PMU és obrir el circuit cada 2 segons. Un cop es troba el circuit obert mesura el voltatge (mesura en quin punt de l'eix de les X es troba) i el torna a tancar. Un cop tancat el circuit sap sobre quina corba es troba i fixa de nou el voltatge del motor, en el punt màxim de la corba mesurada, és a dir, en el punt on obtindrem el màxim rendiment de potència del nostre generador.

5 OBTENCIÓ DE VOLTATGE DE L'ENERGY HARVESTING

Per obtenir el voltatge que ens proporciona l'EH utilitzarem el mètode infraroig, explicat en el capítol 4.3.6.

Durant el projecte d'enginyeria mecànica es va utilitzar un engranatge i se li va fer un forat per tal de mesurar el parell motor. Utilitzarem aquest mateix engranatge amb el forat per tal de mesurar el temps que tarda en donar una volta.

Això o farem col·locant el fotoemissor en un costat de l'engranatge, i el fototransistor en l'altre, de manera que al passar el forat de l'engranatge entre l'emissor i el receptor obtindrem un pic.

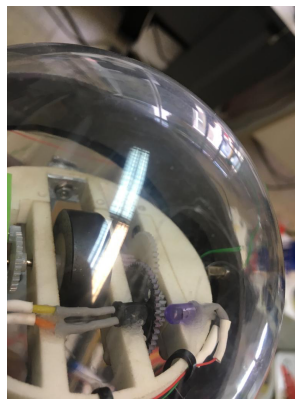


Figura 5.1. Engranatge amb el fotosensor.

Un cop tinguem la gràfica de l'assaig, aplicarem la transformada de Fourier per poder veure el mapa de freqüències i d'aquesta manera poder obtenir a quina velocitat gira l'engranatge i per tant, l'eix del generador.

Hem de tenir en compte que només hem introduït el tren d'engranatges de la part superior.

Per tant, el tren d'engranatges que utilitzarem durant el test, serà el de la següent imatge.

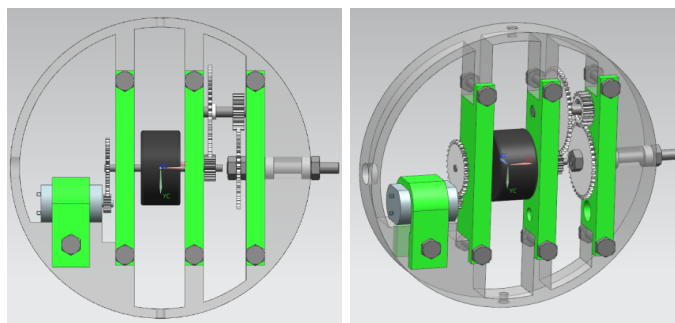


Figura 5.2. Engranatges Energy Harvesting.

El tren d'engranatges que podem veure en la imatge anterior, equival a:

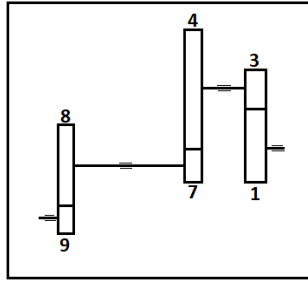


Figura 5.3. Càlcul de la relació de transmissió.

$$i = -\frac{z_1}{z_3} * \frac{z_4}{z_7} * \frac{z_8}{z_9} = -\frac{42}{18} * \frac{48}{12} * \frac{30}{8} = -35$$

Per tant, augmentem la velocitat de l'eix del motor, en referencia al primer engranatge de 35 vegades, en el nostre cas, el fotoemissor i el fotoreceptor l'hem introduït en el engranatge número 4, per tant, al captar els valors aquest engranatge ja ve accelerat 42/18, valor que hem d'obviar, en la nostre equació, finalment tenim que la relació d'augment és de:

$$i = \frac{-z_4}{z_7} * \frac{z_8}{z_9} = -\frac{48}{12} * \frac{30}{8} = -15$$

Amb la transformada de Fourier que hem vist al llarg del punt 3.6, amb els sensors introduïts en l'engranatge número 4 i el seu mapa de freqüències, podem saber la freqüència de gir de l'engranatge nombrat anteriorment, ara és el moment de passar-ho a revolucions per minuts amb la formula següent:

$$\frac{rev}{min} = freqüència de gir * velocitat d'augment * 60 = freqüència de gir * 15 * 60$$

Un cop sabem les revolucions per minut és el moment de parlar de potencia.

El departament de SARTI, en el centre tecnològic de la UPC de Vilanova i la Geltrú, ha fet una caracterització del generador en relació revolucions/minut amb la potència que obtenim. Aquesta caracterització i l'article el podem veure sencer adjunt a al biografia d'aquest mateix projecte, els resultats es mostren en la imatge 4.14.

A partir d'aquesta gràfica podem determinar en funció del rang en que ens movem (una corba o una altre) en rev/minut i la sortida en potència que obtindrem en el motor.

Ara ja sabem com obtenim i quina potencia ens donarà el nostre generador en funció de les revolucions per minut, i per tant, com hem pogut veure, depenent de la freqüència que gira l'engranatge el qual obtenim les dades.

Per acabar, en l'apartat d'anàlisi dels resultats, com a dada de curiositat obtindrem el % de rendiment de la placa ADP5092, amb la següent gràfica:

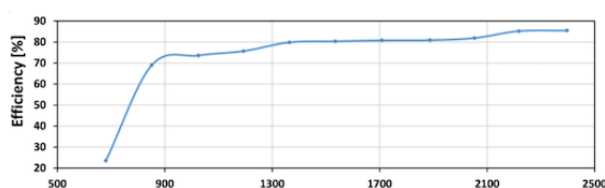


Figura 5.4. Rendiment del AD5092. [Font: Matias Carandell].

6. SOFTWARE DEL DISSENY

6.1 DIAGRAMES DE BLOCS

A continuació mostrarem els diagrames de blocs del node ESP i de Labview.

6.1.1 DIAGRAMA DE BLOCS NODE MCU ESP8862

Degut a les dimensions del diagrama de blocs del node s'ha dividit en dos parts.

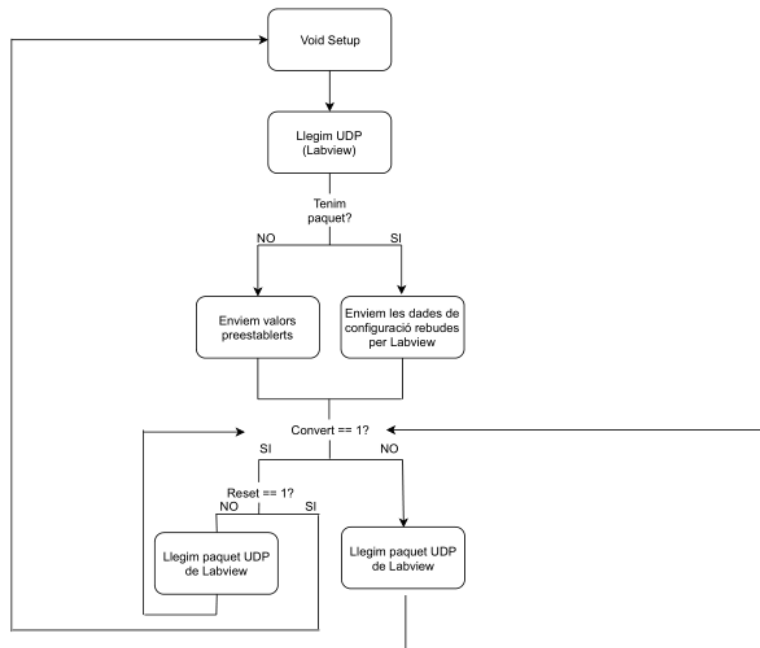


Figura 6.1. Diagrama de blocs ESP part 1.

El primer que fem al iniciar el node és crear i assignar totes les variables, llibreries, inicialitzacions dels pins, baudrets, connexió wifi, etc que utilitzarem.

Un cop inicialitzades totes les comandes necessàries les quals veurem mes endavant, entrem en el void loop, un bucle que es repetirà al llarg de l'estada en marxa del node.

En el void loop, el primer que ens trobem és amb un condicional, en aquest primer condicional hi ha una variable que fa que només s'executi un cop cada vegada que el node encengui, ja sigui per primer cop o al resetejar. Serveix per enviar les comandes pertinents al TD1205P i configurar-lo a la nostre elecció.

Llegim si tenim paquet per UDP, provinent de Labview, és a dir, mirem si tenim un paquet enviat des de l'ordinador. Si el tenim, vol dir que des de Labview hem assignat la freqüència i l'escala a la que volem que transmeti el TD1205P, per tant el que farem és enviar aquest paquet de dades al TD1205P per que s'iniciï amb la configuració que nosaltres volem. Per seguretat, o per si ens està bé amb la configuració preestablerta, (Freqüència = 25Hz i escala 1G) s'ha creat el condicional de si no hi ha paquet, en aquest cas enviarem les comandes preestablertes.

A continuació ens trobem amb una variable anomenada convert, és la variable escollida per dur a terme el reset, és a dir, si és igual a 1, el TD1205P és reiniciarà, si és 0, el que farem és seguir amb el codi, llegirem si tenim paquet UDP altre cop, ja que, en aquest paquet enviat des de Labview, tenim altres variables, que podem enviar a UBIDOTS o no.

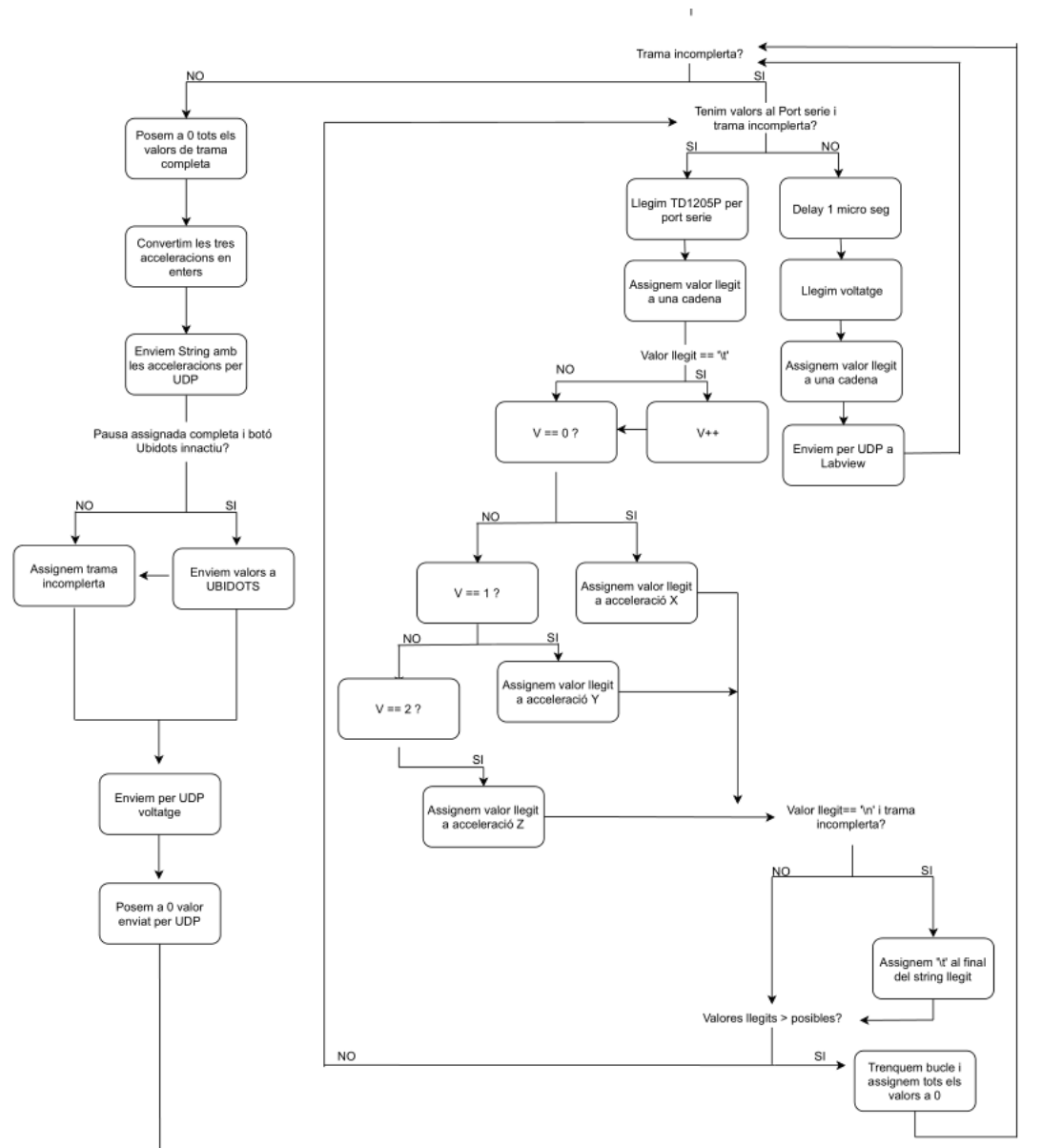


Figura 6.2. Diagrama de blocs ESP part 2.

Un cop hem llegit les variables (en cas que el buffer les tingui) anem a comprovar si tenim la trama completa, és evident que el primer cop no la tindrem, per tant entrarem, i mirarem si tenim valors en el buffer del port sèrie, en el cas que no tinguem valors el que farem, ja que la trama no és complerta, és llegir els valors que ens proporciona l'energy harvesting. Enviem aquests valors per UDP a Labview i tornarem a preguntar si tenim valors, així continuament fins que el nostre TD1205P envii les dades.

Si tenim valor pel port sèrie, el que farem és llegir aquest valor, l'assignarem a una cadena i a una variable, veurem mitjançant unes condicions a quina variable pertany, acceleració de X, de Y o de Z. Aquestes variables son utilitzades mes endavant per poder enviar a Ubidots.

A continuació comprovarem que no hem rebut la comanda de final de trama (\n) i en cas negatiu tornarem al principi de trama completa, així consecutivament fins que acabem de rebre tots els caràcters de les tres acceleracions, moment en que rebrem el caràcter fi de trama, \n, en aquest moment al final d'aquest bucle li assignarem a la cadena \t, per separar mes endavant amb Labview els valors de manera més còmode. Per acabar amb aquest bucle comprovarem si el valor llegit és inferior al valor màxim que podem rebre, ja que si el valor rebut és mes gran que el permès, vol dir que hem perdut la trama per algun error, com per exemple adjuntar dos trames seguides, perdent el valor \n de fi de trama, en cas que el valor sigui mes gran, trencarem el bucle, assignarem tots els valors a 0 i tornarem a començar a llegir valors.

Un cop hem rebut tota la trama, ja no entrarem en el bucle de trama incompleta, ja que al estar completa anirem al costat negatiu. El que farem en aquesta part del codi, és assignar tots els valors de trama completa a 0, d'aquesta manera a la següent volta podrà tornar a entrar, convertirem les 3 variables guardades en enters, per poder enviar a Ubidots. A continuació enviem la cadena de valors llegida per UDP a Labview per poder tractar-los. Finalment comprovem si ha passat la pausa assignada per enviar a Ubidots, en cas que hagi passat i des de Labview estigui la opció activa d'enviar, enviarem els valors, assignem la trama com a incompleta i netegem la cadena de valors enviada posant-la a 0 per evitar errors en la següent lectura, d'aquesta manera, tornarem a començar tot el procés.

6.1.2 DIAGRAMA DE BLOCS LABVIEW

El programa Labview és un entorn de programació el qual es programa mitjançant blocs, degut a les dimensions del diagrama, el dividirem en parts. Cal entendre que totes les parts es troben dins d'un loop, és a dir, es repeteixen contínuament cada cop que s'executa el programa.

En primer lloc tenim el que s'envia des de Labview cap al node ESP.

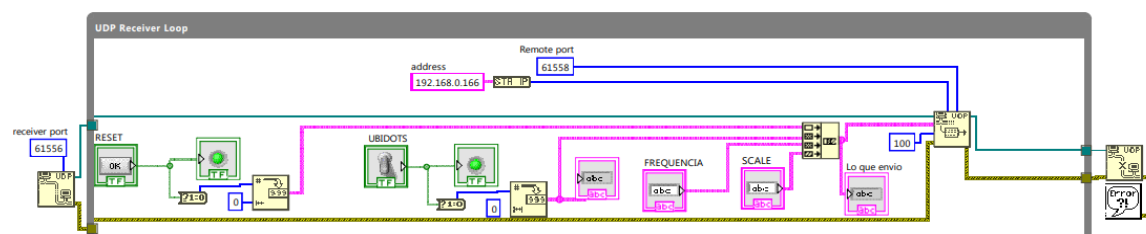


Figura 6.3. Diagrama de blocs Labview part 1.

Abans d'entrar en el loop, tenim el UDP open Function el qual s'encarrega d'iniciar la transmissió UDP, en aquest bloc se li ha d'assignar el port el qual rep informació, en el nostre cas es indiferent quin port li assignem, ja que, per aquest no rebem res.

A continuació tenim 4 opcions de control, que els governa el propi Labview mentre es troba connectat amb el node, aquestes opcions son:

- Reset: boto amb el que executa un reset al node, amb aquesta opció trobem un led que ens avisa el moment en que executem el reset.
- Ubidots: aquesta opció ens permet habilitar o des habilitar l'enviament de dades a Ubidots, degut als problemes que veurem mes endavant, és una opció molt valida per poder enviar i rebre dades amb mes velocitat. Aquesta opció de control també té incorporat un led per avisar si es troba en estat ON o OFF.

- Per acabar, tenim dos opcions que ens permeten escriure des de l'entorn Labview. Fan referencia a la freqüència i a l'escala que volem que envii informació el TD1205P, cap al Node. Cal comentar que, com hem dit en l'apartat anterior, enviem les dades una vegada cada cop que s'encén el node, per tant al variar la freqüència o l'escala no aplicarà el valor al TD1205P fins que no fem un posterior reset.

A continuació adjuntem aquets 4 paràmetres en un String i l'enviem al node mitjançant el bloc UDP Write Function que li hem de passar dos paràmetres: en primer lloc el port on volem enviar la informació, i en segon lloc l'adreça IP a la que es troba connectat el nostre Node.

El bloc que podem veure a la part dreta fora del loop, només s'executa un cop, quan parem la comunicació de Labview, i serveix per tancar la comunicació UDP.

A continuació trobem el loop en el que tractem les dades.

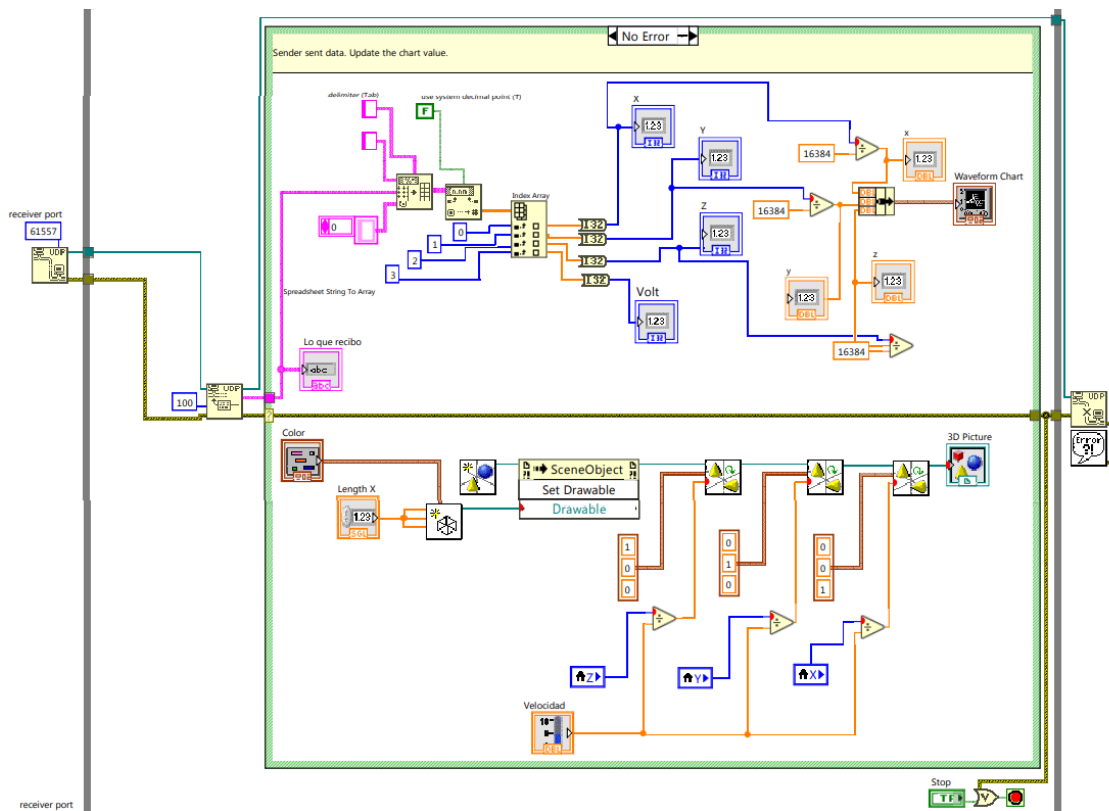


Figura 6.4. Diagrama de blocs Labview part 2.

En primer lloc trobem fora del loop el bloc UDP open Function, que hem explicat abans, la diferencia és que en aquest cas si que és important el port, ja que rebem informació provinent del Node ESP.

El següent bloc que trobem abans del Case structure (requadre que podem veure de color verd), serveix per llegir la informació que rebem pel port assignat anteriorment. En aquest bloc li hem assignat una pausa de 100 milisegons per no trobar errors en la lectura.

El Case structure, serveix per programar diferents casos que ens podem trobar en diferents situacions, en el nostre cas hi ha programat dos casos, si hi ha error en la lectura i si no hi ha error.

Si no hi ha error en la lectura el primer que fem és introduir les dades llegides en una matriu amb la funció Spreadsheet String to array function, per poder separar-les més fàcilment amb el següent bloc, el caràcter que indica la separació és \t.

Un cop tenim els valors separats, mostrem per pantalla (Labview) els valors llegits (valors bruts) i els passem de caràcter a enter, com hem dit anteriorment aquests valors s'han de tractar, per aquest motiu dividim cada una de les acceleracions separades anteriorment per el valor màxim que podem obtenir amb el TD1205P, d'aquesta manera tenim la informació llegida en G's. Per acabar, mostrem els valors en G's per pantalla i els introduïm en un gràfic que anirà avançant a mesura que avança el temps.

Podem veure que a part de l'acceleració, també separem un valor anomenat Volt, aquest valor és per si volem enviar el voltatge llegit per el Node juntament amb la cadena o per si el volem enviar per separat, depenent si enviem paquets de dades o no. En el nostre cas, amb la programació que s'ha fet en el Node, tenim deshabilitada aquesta opció, ja que l'enviem per un altre port, així que aquest valor serà nul.

Les acceleracions llegides les convertim en variables locals per poder tractar-les en aquest mateix case structure.

Un cop hem llegit les acceleracions, trobem un conjunt de blocs els quals ens formen un cub, que és mou exactament igual com es mogui el TD1205P, és a dir, tal qual és mogui la boia. Per dur a terme aquest procés, creem un color i unes dimensions, creem l'objecte, un cub en el nostre cas, i l'introduïm cada una de les acceleracions llegides (per això s'han assignat com a variables locals anteriorment) en cada un dels eixos cartesianes que ens permet introduir la funció per poder moure el cub en funció és mogui la boia.

Cal comentar que abans d'introduir els valors de les acceleracions en els eixos, els dividim per un valor, aquest valor l'escollim en l'entorn de Labview i serveix per que els valors de les acceleracions llegides son molt elevats, i el cub és mou molt ràpid, amb aquest slider introduït, podem fer que el cub no tingui moviments tan prolongats, el valor adequat per poder veure bé els moviments del cub és troba entre els 9000 i els 10000.

Per acabar, igual que en el cas anterior, executem el bloc per tancar la comunicació UDP, només en el moment en que parem el programa Labview.

Com hem dit, hem programat dos possibles casos, un primer que acabem d'explicar si no trobem error en la lectura, i un segon si trobem error, el qual el podem veure a continuació.

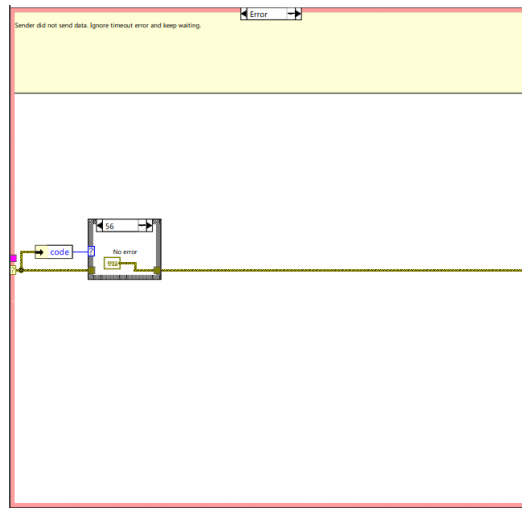


Figura 6.5. Diagrama de blocs Labview part 3.

Aquest case structure simplement ens executa un codi d'error, el numero 56, el qual ens diu que s'ha llegit malament els valors o que hi ha hagut algun error en la comunicació durant la lectura.

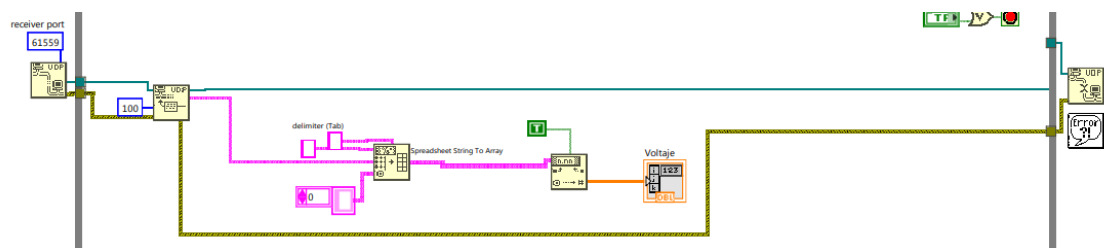


Figura 6.6. Diagrama de blocs Labview part 4.

El port per el que llegim el voltatge enviat per el Node, és el 61559, amb els dos mateixos blocs per obrir la comunicació i per llegir-la, amb una pausa programada com el cas anterior de 100 mili segons.

El valor que llegim per el Node, l'enviem amb la finalització \t, d'aquesta manera igual que en el cas de les acceleracions separem els valors llegits per aquest caràcter, i només ens quedem amb els valors que trobem abans del \t, descartant si trobem algun valor erroni durant la comunicació, finalment mostrem el valor llegit en l'entorn Labview, i tanquem la comunicació com hem explicat en els casos anteriors.

Per acabar aquest loop, el que fem és guardar els valors llegits en un fitxer .csv, que podem obrir amb el mateix Excel.

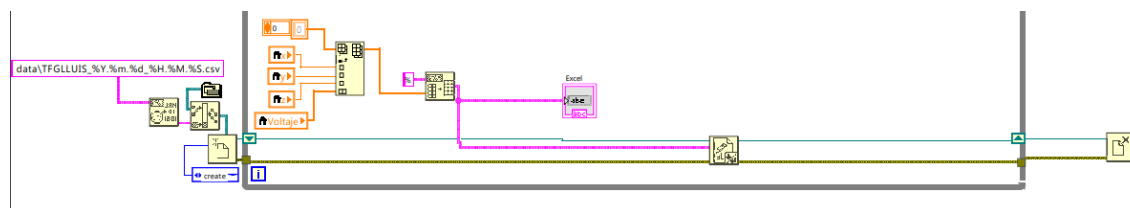


Figura 6.7. Diagrama de blocs Labview part 5.

Amb aquests conjunt de blocs el que fem és crear un fitxer dins de la carpeta anomenada data, aquest fitxer portarà el nom de "TFGLUIS_any, mes, dia_hores, minuts i segons" els quals en

els que s'ha creat aquest fitxer. Adjuntem els valors en un vector, i els escrivim en l'arxiu .csv, per acabar tanquem el fitxer, un cop sortim del loop, és a dir, un cop aturem Labview.

Com a resultat final d'aquest diagrama de blocs obtenim el següent display, amb totes les opcions disponibles mencionades anteriorment per poder variar els paràmetres nombrats.

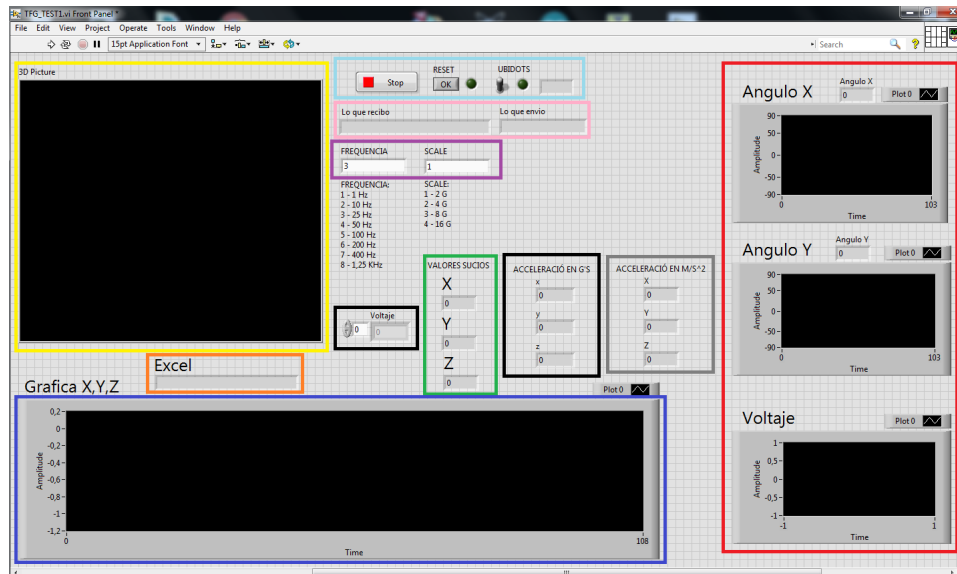


Figura 6.8.Display Labview.

Requadre de color verd: Son els valors “bruts” que obtenim del TD1205P a través del node ESP.

Requadre de color groc: Podem veure el cub exactament com és mou la boia.

Requadre negre: Voltatge enviat per UDP per el port 61559 i valors correctes en G de l'acceleració.

Requadre de color blau fort: Gràfic on podem veure els 3 valors de l'acceleració.

Requadre de color blau clar: Botó stop, per aturar el programa, tancar el fitxer excel i la comunicació UDP. Botó reset, per reiniciar el node (al canviar de freqüència, escala, etc.). Botó Ubidots, per para l'enviament de dades cap a ubidots.

Requadre de color rosa: Dades que rebem, tal i com les rebem en una cadena i la cadena de valors que enviem.

Requadre de color lila: Freqüències i escales a enviar i quadre de text per introduir el valor a enviar.

Requadre de color Vermell: Gràfica dels angles referenciats als eixos cartesianes del TD1205P.

Requadre de color negre: Valors de les acceleracions en G's.

Requadre de color girs: Valors de les acceleracions en m/s².

Requadre de color taronja: Valors emmagatzemats al fitxer Excel.

6.2 LLIBRERIES UTILITZADES

Al llarg d'aquest apartat explicarem les llibreries utilitzades del node ESP8266.

Una dels avantatges que té programar amb arduino, és que ve amb moltíssimes llibreries, aquestes venen amb funcions programades que estudiant el seu funcionament faciliten molt la feina a l'hora de programar.

Les llibreries utilitzades al llarg d'aquest projecte han estat:

- `SoftwareSerial.h`: Aquesta s'utilitza com podem intuir per el propi nom per la comunicació sèrie, en els pins Rx i Tx, també incorporada per USB. Com hem mencionat anteriorment el suport es realitza a través d'una peça de hardware anomenada UART. Aquest chip permet que el nostre node rebí informació inclús mentre treballa en altres tasques, sempre que hi hagi espai en el buffer de 64 bytes.

Aquesta llibreria s'ha programat per que permeti la comunicació des d'altres pins digitals, utilitzant un software per replicar la funcionalitat.

Al llarg d'aquest projecte s'ha utilitzat aquest port sèrie virtual, per mostrar per pantalla a part de Labview mitjançant el programa Tera term, i el cable TTL mencionat anteriorment per poder veure si els valors que rebíem i els que enviàvem eren els correctes amb una velocitat màxima de 115200 baudis.

Una limitació que té aquesta llibreria és que si s'utilitzen diferents ports sèrie a l'hora, només un d'ells pot rebre dades. A l'hora de pujar el programa al node per exemple, s'ha de desconnectar el cable de terra amb el TD1205P, ja que sinó tenim error al pujar el programa, perquè detecta que estem utilitzant l'altre port sèrie. Els pins utilitzats per dur a terme aquest port sèrie virtual han estat el D1 i el D2 del Node ESP.

Les funcions utilitzades d'aquesta llibreria son totes les que ens serveixen per enviar per port sèrie al TD1205P:

- `SoftwareSerial swSer(pinRx, pinTx, false, 256);` per crear el port virtual.
- `Serial.begin(BAUD_RATE);` per declarar els baudis.
- `Serial.println("AT$ACCDATA=1");` per enviar els paràmetres de configuració.
- `ESP8266Wifi.h`: Aquesta llibreria és l'encarregada de la connexió Wifi, amb una sèrie de comandes les quals veurem mes endavant, podrem connectar-nos a internet amb una extrema facilitat, de la mateixa manera, podrem saber quina direcció IP tenim, on ens connectem, si estem connectats i moltíssimes ordres més que podem saber nombrant una simple funció.
- `WiFi.begin(ssid, password);` la utilitzem per connectar-nos al Wifi.
- `WiFi.status();` per saber si estem connectats o no.
- `WiFi.localIP();` amb aquesta funció sabem quina és la direcció IP del ESP.
- `WifiUDP.h`: Així com la llibreria anterior ens permet la connexió wifi, aquesta altre ens permet enviar i rebre paquets per UDP. Els paquets UDP que surten és troben limitats a 72 bytes, per totes les plaques que son AVR, per plaques que no son AVR el límit és fins a 1446 bytes.

En el node ESP, la placa té un AVR (Automatic voltatge regulator), que és un dispositiu hardware que s'utilitza per mantenir el voltatge, per tant, el màxim que podem enviar per UDP és 72 bytes.

- `Udp.begin(localUdpPort);` Iniciem la comunicació per enviar dades.
- `Udp.beginPacket(IP_Enviar, 61557);` Assignem a quina IP i quin port enviarem les dades.
- `Udp.write("New conection.");` Per escriure un conjunt de caràcters o els valors rebuts per el TD1205P.
- `Udp.endPacket();` Per finalitzar la comunicació UDP després d'enviar un paquet.
- `Udp.parsePacket();` En aquesta variable hi ha el valor que obtenim de la lectura UDP.
- `Udp.read(incomingPacket, 255);` Serveix per llegir els valors rebuts per UDP.

- `UbidotsMicroESP8266.h`: Aquesta llibreria ens permet enviar i rebre paràmetres a Ubidots. El principal problema d'aquesta llibreria, és que necessita molt temps per enviar les dades. Es una llibreria perfecte per poder rebre dades en temps real, sempre i quant aquestes dades no canviïn molt ràpidament. Seria la llibreria perfecte per poder saber la localització de la boia, ja que ho podem enviar en espais de temps mes elevats no com en el cas de l'acceleració que varia en mil·lèsimes de segon.

- `client.wifiConnection(WIFISSID, PASSWORD);` Per connectar-nos a Ubidots.
- `client.add(X, xa);` Adjuntem en un caràcter els valors que volem enviar.
- `client.sendAll(false);` Enviar els valors desitjats a Ubidots.

6.3 ENTORN UBIDOTS

A continuació explicarem la comunicació UBIDOTS. Com ja hem dit, és una aplicació gratuïta per veure les dades enviades in situ.

El problema que ens hem trobat, és que no és suficientment ràpid per la comunicació que nosaltres necessitem, tot i que s'ha deixat per treballs futurs, per exemple poder enviar la localització, ja que la localització no la rebem cada 40 mili segons, son períodes de temps molt mes prolongats.

Per poder connectar-nos a UBIDOTS necessitem configurar els següents paràmetres tan a la pàgina Web com al node ESP.

`#define DEVICE "esp8266_lluis"` Primer de tot hem de declarar com és diu el nostre dispositiu. A continuació totes les variables que enviarem, en el nostre cas és el valor de reset, les tres acceleracions i el valor llegit del voltatge.

```
#define VARIABLE "rest" // Per enviar reset
#define X "5caf84fc1d84727829006cdc" //Per enviar acceleració X
#define Y "5caf85021d84727829006cdd" //Per enviar acceleració X
#define Z "5caf85061d847277c6e065c2" //Per enviar acceleració X
#define VOLT "5caf85a81d8472797a02268f" //Para enviar el voltatge
```

Per acabar hem de declarar el token, que és una sèrie de caràcters que obtenim de Ubidots per poder sincronitzar el nostre dispositiu amb la pàgina web.

```
#define TOKEN "BBFF-sHKSpPE18pPebhy3ygLxcvx23n72NY"
```

7 ANÀLISI ANTERIOR AL TEST

En aquest capítol explicarem el hardware utilitzat a l'hora de la programació, així com un seguit de test que s'ha realitzat als aparells utilitzats.

7.1 HARDWARE DEL ANÀLISI

El hardware utilitzat al llarg del anàlisi el podem veure en la següent imatge:

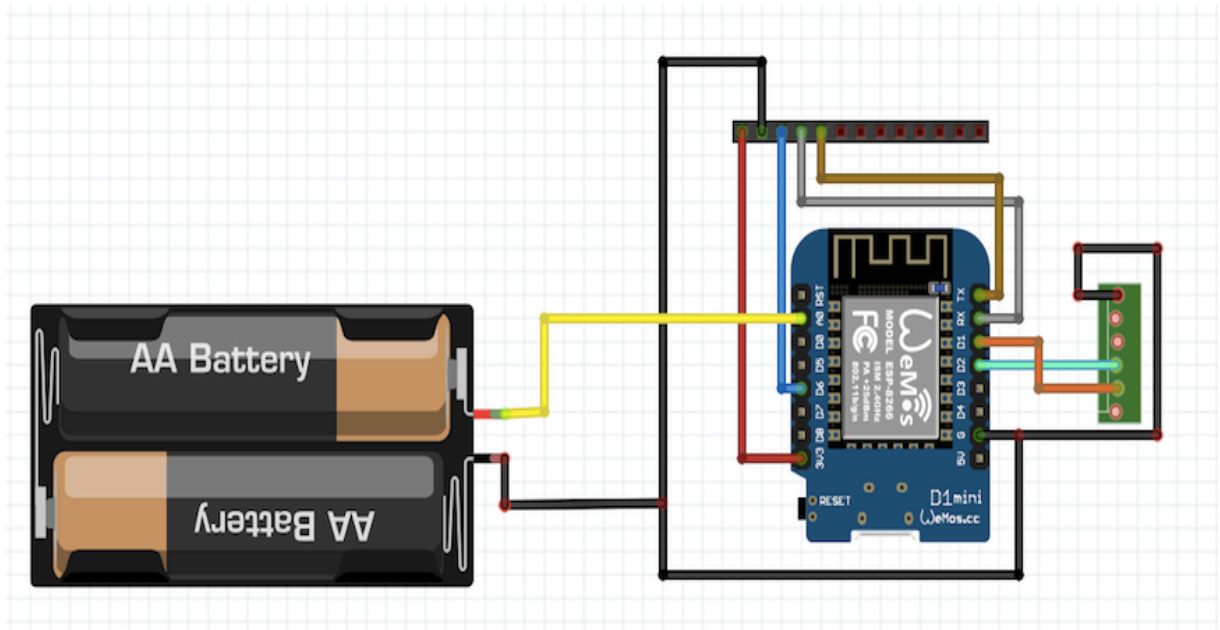


Figura 7.1. Hardware proves.

El connector de color negre representa els connectors de TD1205P, els quals els hem vist en l'apartat 4.2.1. Els dos primers pins els utilitzarem per alimentar el TD1205P a través del ESP8266, el següent pin (cable de color blau) és l'utilitzat per fer el reset des de l'entorn Labview, per acabar tenim els dos connectors del port sèrie RX i TX.

En aquesta transmissió trobem una velocitat de 460800 baudis, i serveix per adquirir, tractar i enviar les acceleracions adquirides amb el TD1205P.

Un problema que ens vam trobar a l'hora de programar, és que no veiem els resultats que obteníem en el ESP, per tant, no podíem saber si els valors que rebíem a Labview eren correctes o no, per aquest motiu, és va comprar un cable TTL 232R-3V3, que és troba representat en la imatge anterior com els connectors de la part de la dreta.

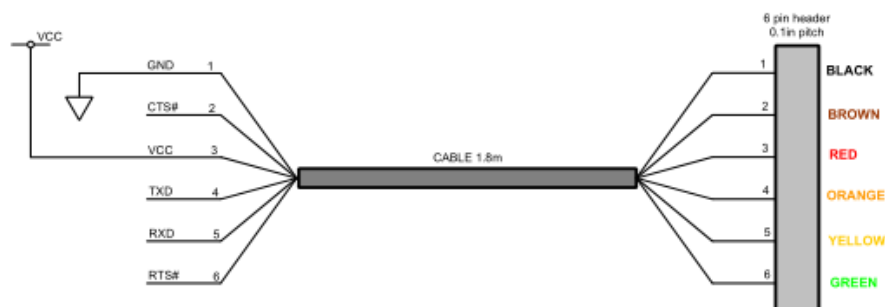


Figure 4.1 TTL-232R-5V and TTL-232R-3V3, 6 Way Header Pin Out

Figura 7.2. Connexió cable TTL. [Font: Datasheet TTI 232R].

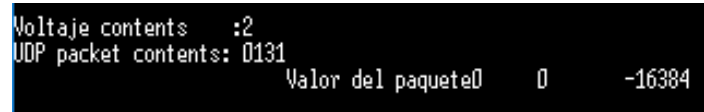
Utilitzant aquest cable, s'ha creat un port fictici en el node ESP8266, d'aquesta manera, mitjançant el programa TERA TERM, podem captar en tot moment què envia el TD1205P, i veure també si el que hem programat s'ha programat be, i que rebem dades coherents. Aquesta connexió va a una velocitat de 115200 baudis.

El port fictici que s'ha creat és troba en els pins D1(TX) i D2 (RX) creuats amb el RX i TX del nostre TTL.

TERATERM, és un programa emulador de terminal (comunicacions) de codi obert, gratuït i implementat per software. Admet connexions de Telnet, SSH 1 i 2 i per acabar el que ens interessa al llarg d'aquest projecte, per port sèrie.

7.2 TEST AMB EL NODE ESP8266 I EL TD1205P

Com hem vist en l'apartat anterior hem creat un port virtual, aquest port virtual ens mostra la informació com:



```
Voltaje contents :2
UDP packet contents: 0131
Valor del paquete0 0 -16384
```

Figura 7.3. Informació port virtual.

La primera línia ens diu el valor llegit per el pin analògic, en aquest cas el valor llegit el tenim a l'aire, i obtenim una mesura de 2mV.

El següent valor ens indica el valor que li enviem des de l'ordinador, des de el programa Labview, la primera xifra equival al reset, sempre que sigui 0 no tindrà cap efecte, serà en el moment en que és posí a 1 quan farà el reset. El següent 1 equival a Ubidots, sempre i quan es mantingui a en estat alt, no enviarem valors a Ubidots. Per acabar tenim dos caràcters mes, la freqüència i l'escala a la que està programada el TD1205P.

Finalment, obtenim els valors de l'acceleració provinents del TD1205P, en aquest cas els valors son: 0 per l'acceleració de les X, 0 per les acceleracions de l'eix Y i el valor màxim (ja que es l'únic eix al que afecta la gravetat) per l'acceleració de l'eix Z.

A continuació mostrarem unes captures de pantalla per poder mostrar el que veiem en Tera Term i en el programa Labview, i podrem analitzar si els resultats obtinguts tenen coherència o no.

Per el TD1205P, en una posició completament recta, obtenim els següents valors per el programa Labview:

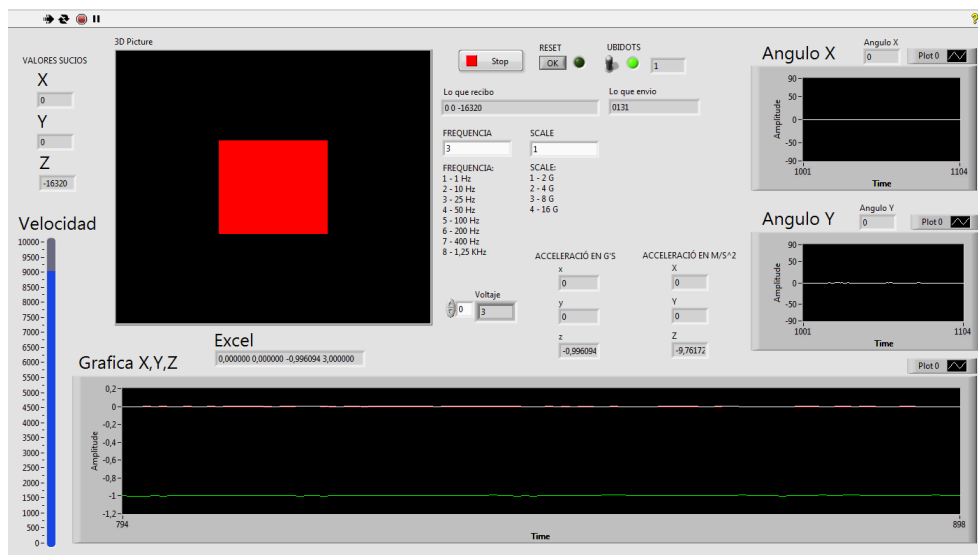


Figura 7.4.Labview TD1205P horizontal.

Com podem veure els valors de l'esquerra son els captats directament des de el ESP sense tractar, els mateixos que obtenim tal i com els rebem, en una cadena al mig del screenshot.

Just a la dreta tenim els valors que enviem 0131, els quals veurem que coincideixen amb els rebuts al nostre node.

Les dos gràfiques de la dreta ens mostren els valors dels angles, en l'eix de les X amb un valor de 0 graus, i el valor de l'angle en l'eix Y de 0 graus.

Finalment podem veure els valors expressats en G's i en m/s^2 , 0 per l'eix de les X, 0 G per el de les Y i 0 G per el de les Z el màxim en el moment captat de 0,996G.

Aquests valors passats a acceleració en unitats de m/s^2 tenim 0 per l'eix de les X y de les Y, mentre que tenim 9,76 m/s^2 en el eix Z, com podem veure aquest valor es quasi el de la força de la gravetat, ja que només afecta a aquest eix.

Els valors que hem obtingut per el programa TeraTerm han estat;

```

Voltaje contents :3
UDP packet contents: 0131
Valor del paquete0 0 -16384

Voltaje contents :3
UDP packet contents: 0131
Valor del paquete0 0 -16320

Voltaje contents :3
UDP packet contents: 0131
Valor del paquete0 0 -16320

Voltaje contents :3
UDP packet contents: 0131
Valor del paquete64 0 -16384

Voltaje contents :3
UDP packet contents: 0131
Valor del paquete0 0 -16384

Voltaje contents :3
UDP packet contents: 0131
Valor del paquete0 0 -16320

Voltaje contents :3
UDP packet contents: 0131
Valor del paquete-64 0 -16320

Voltaje contents :2
UDP packet contents: 0131
Valor del paquete64 0 -16320

```

Figura 7.5.TeraTerm horitzontal.

Els valors obtinguts són els esperats, ja que amb el TD1205P obtenim una mesura quasi nul·la per els eixos X, Y, i un valor màxim per l'eix Z, en l'instant en que els valors de X i Y són 0, l'acceleració de l'eix Z agafa el valor màxim.

Si girem el TD1205P en posició horitzontal cap a l'esquerra obtenim els següents resultats per Labview:

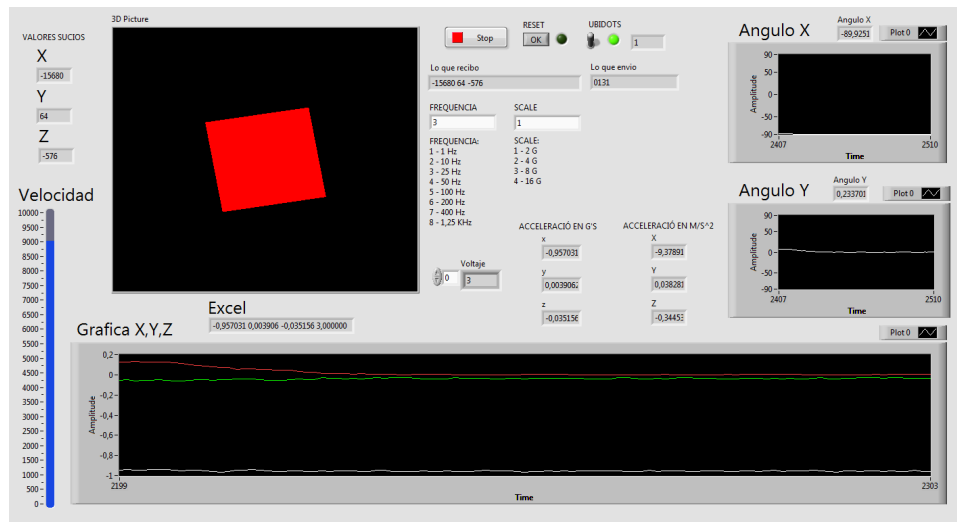


Figura 7.6. Labview 90 graus horitzontal.

Veiem que en aquest cas obtenim un valor quasi nul per l'eix de les Y i per l'eix de les Z, amb un valor per l'eix de les X de 89,92, sent el resultat esperat de 90 graus. Això és degut a que com podem observar, ni l'eix de les X ni el de les Z és nul, si aquests dos valors fossin nuls, obtindríem els 90 graus per l'angle respecte l'eix Y.

El mateix passa amb l'acceleració, tenim una acceleració de 9,37 m/s² per l'eix de les X, i un valor quasi nul per els altres dos eixos.

Finalment, girant el TD1205P 90 graus en direcció de l'eix Y, obtenim els següents resultats:

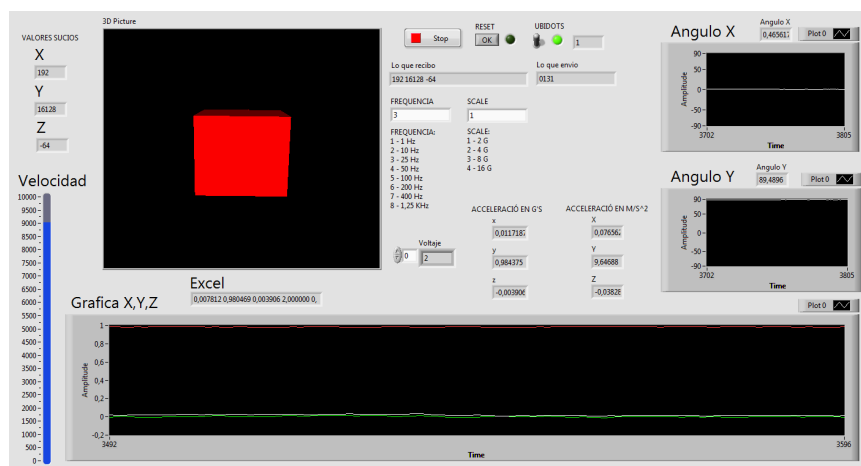


Figura 7.7. Labview 90 graus horitzontal.

Com podíem preveure, en aquest cas, obtenim un valor quasi nul per l'eix X, i per l'eix Z, mentre que per l'eix Y obtenim un valor de 89,48 graus, i una acceleració de 9,64 m/s².

Els valors obtinguts en el Tera Term, han estat els esperats també.

```

Voltaje contents :2
UDP packet contents: 0131
Valor del paquete320 16128 -192

Voltaje contents :2
UDP packet contents: 0131
Valor del paquete320 16128 -64

Voltaje contents :2
UDP packet contents: 0131
Valor del paquete192 16064 -64

Voltaje contents :2
UDP packet contents: 0131
Valor del paquete192 16064 -128

Voltaje contents :2
UDP packet contents: 0131
Valor del paquete320 16064 -128

Voltaje contents :3
UDP packet contents: 0131
Valor del paquete256 16128 -128

Voltaje contents :2
UDP packet contents: 0131
Valor del paquete192 16064 -192

```

Figura 7.8.TeraTerm 90 graus horitzontal.

En aquest cas no hem aconseguit posar el TD1205P del tot recte, tot i que tenim el valor quasi màxim en l'eix de les Y.

7.4 TEST DE LA FREQUENCIA

De la mateixa manera, hem realitzat probes amb la freqüència. Aquestes probes s'han fet mitjançant la funció millis del node MCU per poder veure què tarda en rebre, compactar i enviar el paquet cap a Labview, a mes, veurem si tenim error, a mesura que anem augmentant la freqüència.

Per una freqüència de 1 Hz els resultats han estat de:

```

El periodo de envio UDP ha sido: 1932
UDP packet contents: 0111

```

Figura 7.14.Resultat freqüència 1 Hz.

El valor de UDP packet, ens mostra: el primer 0 vol dir que reset estava desactivat, el segon número, el 1 vol dir que UBIDOTS també estava desactivat, és a dir, no enviem dades a la web, i per acabar els dos següents números son els valors de la freqüència i l'escala a la que és troba programat el TD1205P.

1 Hz = 1, 10 Hz = 2, 25Hz = 3, 50 Hz = 4.

Per tant amb una freqüència = 1 Hz, obtenim valors cada 1932 mili segons, és a dir, cada 1,932 segons.

Per una freqüència de 10 Hz hem obtingut els resultats següents:

```

El periodo de envio UDP ha sido: 160
UDP packet contents: 0121

```

Figura 7.15.Resultat freqüència 10 Hz.

En aquest cas, amb una freqüència del TD1205P de 10 Hz obtenim valors cada 160 mili segons, cada 0,163 segons.

Per una freqüència de 25 Hz hem obtingut els resultats següents:

```
El periodo de envio UDP ha sido: 54
UDP packet contents: 0131
```

Figura 7.16.Resultat freqüència 25 Hz.

Per una programació del TD1205P de 25 Hz obtenim un paquet cada 54 mili segons, cada 0,054 segons.

Per una freqüència de 50 Hz hem obtingut que:

```
El periodo de envio UDP ha sido: 19
UDP packet contents: 0141
```

Figura 7.17.Resultat freqüència 50 Hz.

Amb una freqüència de 50 Hz, obtenim un paquet cada 0,019 milisegons, en aquest cas veiem que la freqüència és massa elevada, com podem veure l'amplitud de l'acceleració de la següent imatge els valors son molt irregulars i per tant erronis.

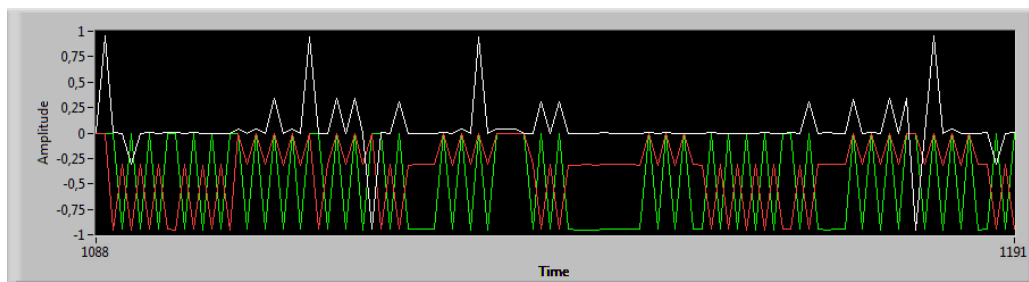


Figura 7.18.Resultat gràfica freqüència 50 Hz.

Per poder enviar a una freqüència superior a 25 Hz, caldria fer paquets mes grans.

Al llarg d'aquest projecte s'ha programat el node ESP8266 de tal manera que, tal i com rep els valors els acumula i els envia un cop té la trama completa. Un cop té els valors de l'acceleració de x, y i z, per poder enviar dades a una freqüència de 50 Hz, caldria acumular una sèrie de paquets en una cadena y enviar tota la cadena amb el conjunt de trames dins.

8 BOIA PER DUR A TERME LES PROVES

Durant aquest apartat explicarem l'estructura que hem muntat per poder tirar la boia al mar i recopilar les dades per poder fer el posterior anàlisi.

En primer lloc hem comprat una bola transparent per amazon, el preu d'aquesta bola és de 11,21€.



Figura 8.1. Bola amb que simularem la boia.

A continuació s'ha posat una plataforma de plàstic, amb un forat en el interior per poder espumar la part de sota de la boia com podem veure en la següent imatge.



Figura 8.2. Espumant la boia.

Després de 18 hores d'espera per que s'endureixi l'espuma, hem enretirat l'espuma sobrant i hem introduït uns pesos de plom per tal que la boia pesi els 3,4 Kg que ha de pesar.



Figura 8.3. Pesos introduïts a la boia.

Aquests pesos s'han col·locat amb una disposició que el centre de masses de la boia queda ben centrat i a una altura no molt elevada, tal i com és pretén que estigui la boia real un cop fabricada.

El resultat de la boia espumada amb tots els pesos introduïts ha estat el següent.



Figura 8.4.Boia espumada amb pesos introduïts.

Just al centre de la plataforma que veiem en la imatge anterior de color gris, hem fet dos forats, per tal de subjectar una protecció per el TD1205P.

En aquests dos forats, hi trobem una femella a l'interior, i una d'altre just tocant la plataforma a la part superior, de tal manera que el cargol roscat que passa pel mig queda fix i centrat en tot moment, finalment amb el TD1205P fix entre els dos cargols roscats, hem col·locat una tapa amb dos forats per que puguin tenir sortida els cargols nombrats anteriorment, roscant una femella just sobre de la tapa queda una estructura molt sòlida impedit cap moviment dels elements del seu interior. L'energy hervesting és col·locarà just al damunt d'aquesta tapa.



Figura 8.5.Fixació TD1205P.

SARTI, ens ha proporcionat un connector de bateria per poder alimentar el node amb una bateria de 9V. En aquest connector hem soldat un connector magnètic normalment obert, d'aquesta manera, podrem interrompre el circuit apropant un imant al connector, i al retirar-lo s'iniciarà el circuit un altre cop, un mètode molt pràctic per no tenir que desmuntar la boia, traient el segellat tota l'estona per tenir que desconnectar la bateria.

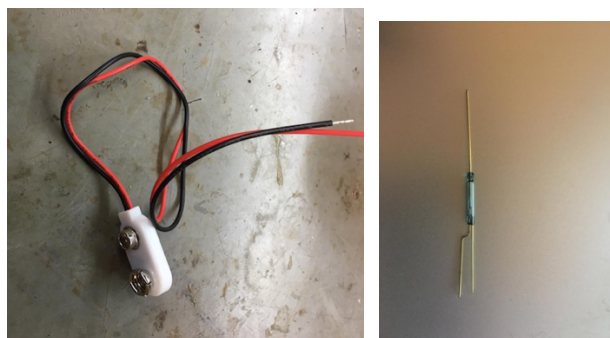


Figura 8.6.Connectors.

Per acabar, hem fixat el node amb 4 claus pels forats destinats a la subjecció i hem col·locat els cables de manera òptima per que no molestin a l'hora d'introduir l'energy harvesting.

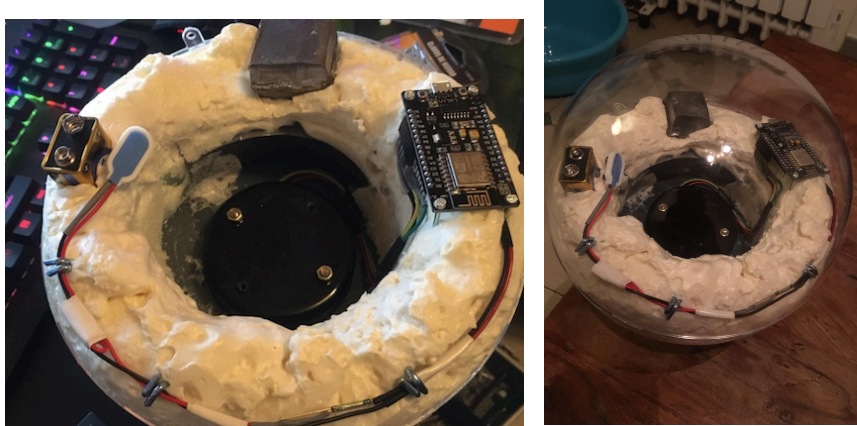


Figura 8.7.Boia amb els connectors i els aparells pertinents.

L'energy harvesting que introduïrem en aquest forat és el de la següent imatge.



Figura 8.8.Energy Harvesting.

Degut a que els resultats els obtenim mitjançant un fotosensor, no obtenim els resultats adequats degut a la llum incident en l'EH, per aquest motiu s'ha tapat tot per tal que els rajos del sol no alterin els resultats.



Figura 8.9.Boia per fer el test.

Per poder veure si l'energy harvesting funciona correctament, no només amb el fotosensor, se li ha incorporat un condensador de 5V, per d'aquesta manera poder veure si movent la boia, es va carregant, aquesta lectura la farem de la mateixa manera que fem la lectura del fotosensor, simplement canviarem el cable que va al pin A0 per tal que llegeixi la càrrega del condensador. Aquest condensador i la PMU ADP5092, la podem veure en la imatge anterior.

Finalment se li va aplicar una cinta per tal d'evitar que l'aigua entrés a l'hora de dur a terme les proves al mar.

Abans de fer el test, es va provar la boia, tant per veure si funcionava correctament el fotosensor com per veure si les dades que rebíem llegint el condensador eren correctes, els resultats obtinguts van ser els següents.

Realitzant la prova llegint el fotosensor:

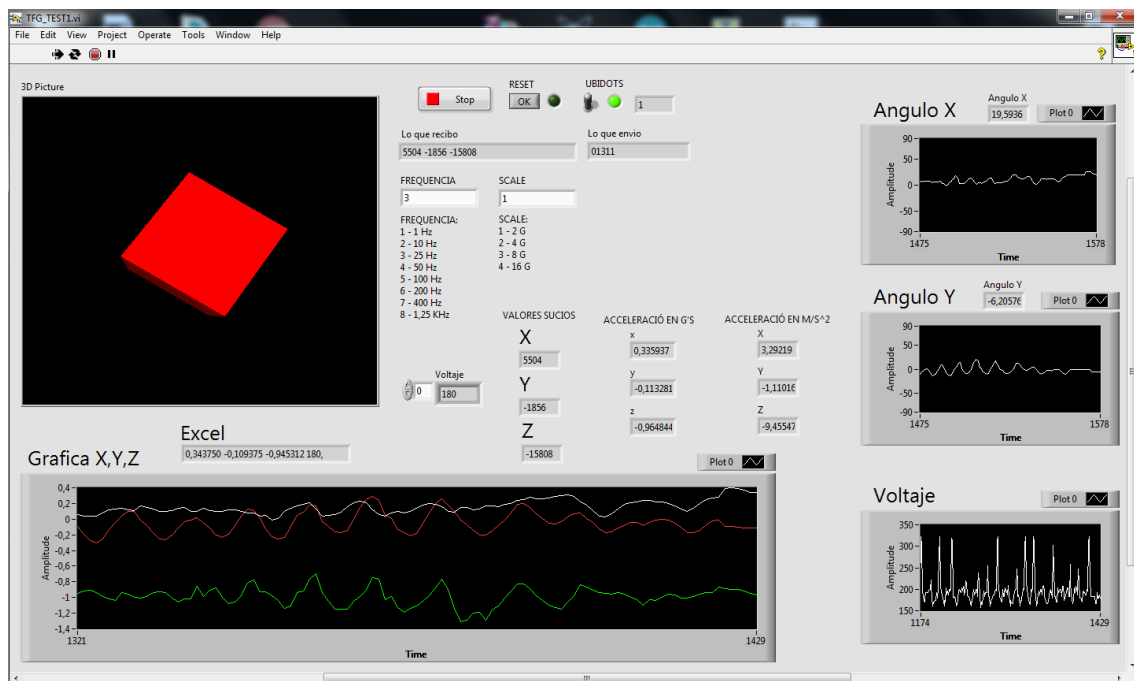


Figura 8.10.Resultats abans de la prova final, fotosensor.

Com podem veure en el gràfic que voltatge, hem obtingut una sèrie temporal, és a dir, un conjunt d'ones simples, el valor llegit en el moment de fer la captura de pantalla és de 180 mV, com podem veure en el display que posa voltatge.

En el cas del condensador, els resultats han estat els següents.

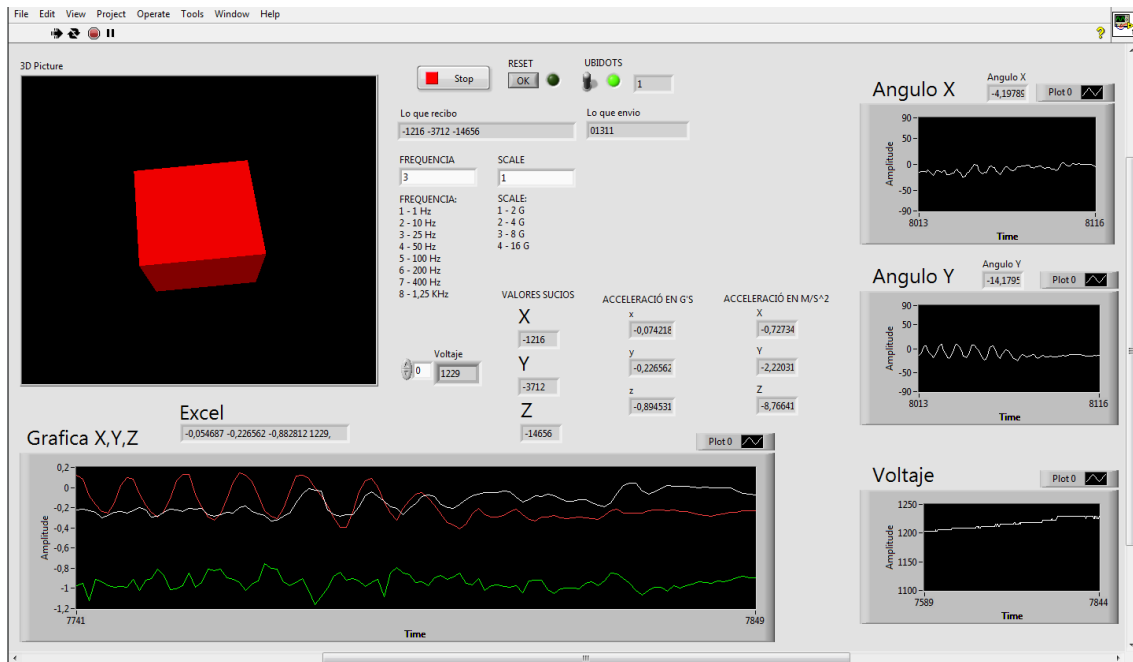


Figura 8.11. Resultats abans de la prova final, condensador.

En aquest cas el valor llegit per el voltatge de display es de 1229 mV, com podem veure en la gràfica del voltatge, comença aproximadament al voltant dels 1200 mV, acabant fins els 1229, per tant, podem dir que el condensador va carregar.

9 METODOLOGIA EXPERIMENTAL

La recopilació de dades es va dur a terme prop de la costa de Sitges, a la platja d'Aigua dolça, coordenades 41°14'10.3"N 1°49'39.8E.

Es va utilitzar una bateria de 12V per alimentar el router.



Figura 9.1. Router i ordinador amb els que es van fer el test al mar.

Juntament amb l'ordinador portàtil per recopilar les dades i emmagatzemar-les al fitxer Excel, és va llençar la boia al mar.



Figura 9.2. Router i ordinador amb els que es van fer el test al mar.



Figura 9.3. Test de la boia al mar.

Un problema que ens vam trobar a l'hora de llençar la boia al mar, es que teníem gaps de dades, és a dir, valors nuls en alguns moment de la lectura, tot i que aquests valors son relativament pocs, ens alteraven els resultats a l'hora d'extreure les dades.

Aquests gaps eren deguts a la connexió Wifi, la boia la teníem massa lluny i no assolí una connexió òptima, perdent dades per el camí.

Per aquest motiu es va tornar a anar al mar per testejar de nou la boia, aquest segon cop amb la precaució de no allunyar massa el wifi de la nostre boia. En aquest segon intent els resultats obtinguts van ser molt mes bons, sense cap gap en les dades i amb uns resultats els quals els podrem veure en el següent capítol.

Cal comentar que els resultats obtinguts durant aquest test, no son els resultats que ens trobarem en mar profund, aquest test és va realitzar molt a prop de la costa, on hi ha un tipus d'ona molt diferent a la que podem trobar a kilòmetres cap endins del mar, tot i que per veure si funciona correctament i si podríem obtenir els resultats a mar obert, és suficient.

10 ANALISI DELS RESULTATS

Un cop tenim els resultats en un fitxer Excel, els introduïrem dins del programa Matlab, per tal de graficar els resultats i treure les transformades de Fourier.

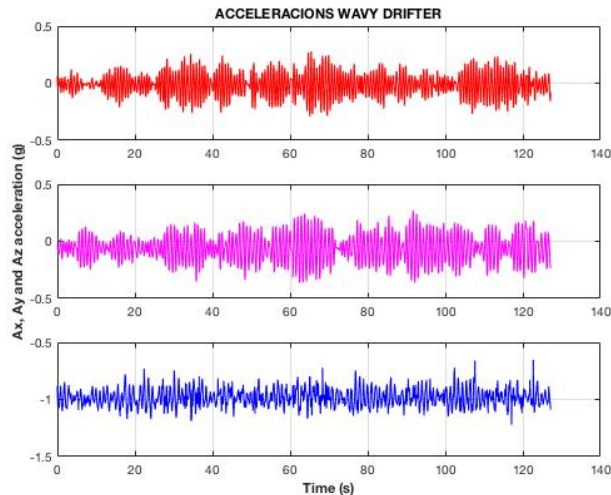


Figura 10.1. Acceleracions del test.

Aquests valors corresponen a les acceleracions de l'eix X gràfic de color vermell, l'eix Y el gràfic de color rosa, i l'eix Z el gràfic de color blau, en cada instant de temps.

Analitzant els valors de l'eix X, veiem que:

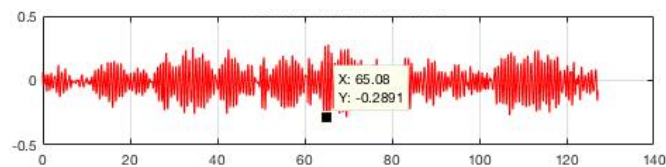


Figura 10.2. Acceleracions de l'eix X.

Per aquest eix veiem que el valor màxim absolut ha estat en el segon 65 de l'assaig, corresponent a $-0,2891G = 2,836m/s^2$.

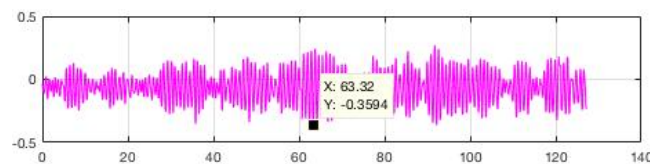


Figura 10.3. Acceleracions de l'eix Y.

En l'eix Y el valor màxim el trobem en l'instant de 63 segons de l'assaig, corresponent a $-0,3594G$, equivalent a $3,52m/s^2$.

Per acabar, analitzarem l'eix Z:

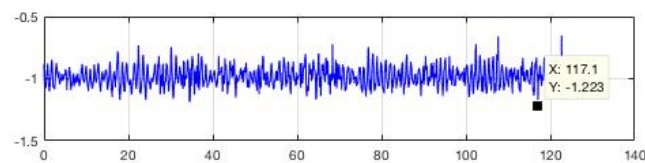


Figura 10.4. Acceleracions de l'eix Z.

Com podem observar, els resultats en aquest eix son molt mes elevats, això es degut a que aquest eix es l'afectat per la força de la gravetat. El valor màxim obtingut al llarg de l'assaig correspon a $-1,223G$, $11,997 \text{ m/s}^2$, per tant, l'acceleració que obté produïda per la ona, ha estat de $2,18 \text{ m/s}^2$.

Analitzant els resultats veiem que el dia de l'assaig no hi havien ones molt elevades, ja que les acceleracions mostrades tenen un valor molt baix, a continuació mostrarem la transformada de Fourier, per poder veure la freqüència d'aquestes acceleracions.

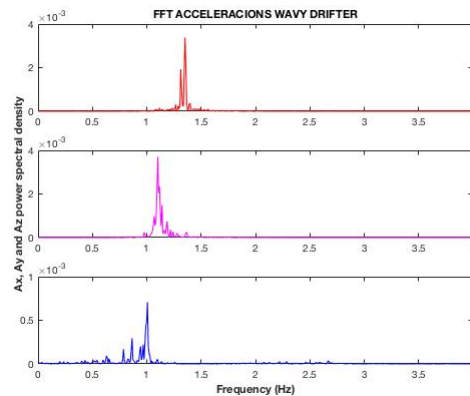


Figura 10.5. Transformada de Fourier de les acceleracions mesurades.

Només observant els gràfics de l'eix X i de l'eix Y, ja podem veure que algun factor ha alterat els nostres resultats, aquests dos eixos no tenen la mateixa freqüència, això no hauria de ser així, degut a que els dos eixos son influenciats per les mateixes ones, i per tant deurien de tenir la mateixa freqüència.

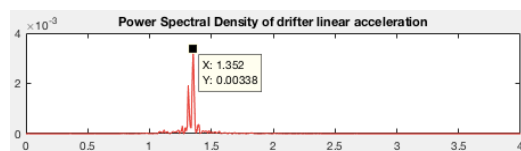


Figura 10.6. Transformada de les acceleracions de l'eix X.

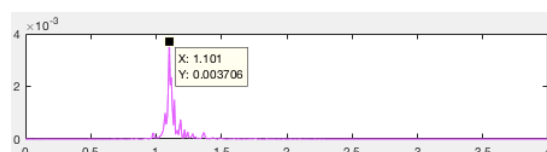


Figura 10.7. Transformada de les acceleracions de l'eix Y.

Això és degut a que, per tal que la boia és mantingui estable en repòs al terra, se li va introduir un pes com el que podem veure en la següent imatge.



Figura 10.8. Pes introduït per equilibrar la boia.

Al introduir aquest pes, al desar la boia al terra és manté estable (amb el TD1205P en posició horitzontal), però a l'hora d'introduir aquest pes vam passar per alt els moments d'inèrcia, alterant els resultats en els eixos Y.

A continuació analitzarem els resultats del fotosensor.

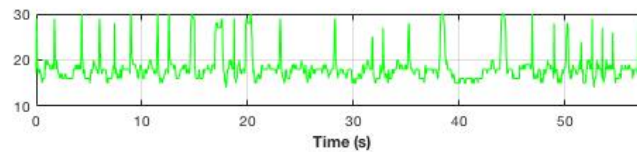


Figura 10.9. Freqüència del fotosensor.

En la gràfica 10.9, podem veure els pics que ens emet els sensors introduïts en l'engranatge, cada un d'aquests es en el moment en que el forat de l'engranatge passa per el sensor infraroig, és a dir, cada un d'aquests pics representa una volta del engranatge.

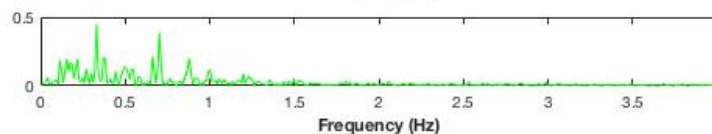


Figura 10.10. Transformada de Fourier del fotosensor.

Podem veure que la transformada de Fourier, es troba entre els 0.4 i 1 Hz.

Aplicant la formula del capítol 5, podem obtenir les revolucions a les que gira l'eix del generador.

$$\frac{rev}{min} = freqüència de gir * velocitat d'augment * 60 = freqüència de gir * 15 * 60$$

$$\frac{rev}{min} = 1 * 15 * 60 = 900 rev/min$$

Si interpretem aquest resultat en la gràfica on es caracteritza el generador, Figura 4.14 que trobem en el capítol 4.3.7, veiem que la potencia que acumularíem, és de 1.3 mW.

Per acabar analitzarem l'eficiència de la PMU, aquest valor el traiem mitjançant les revolucions a la que gira l'eix del generador amb la Figura 5.4.

L'eficiència que trobem del AD5092 és de 74%.

Com hem dit aquest es un resultat molt inferior al esperat, això és degut a que els dies que és van realitzar els tests, el nivell de les ones no era l'adequat, era molt inferior.

D'altre banda, les ones que ens trobem a prop de la costa, solen ser ones baixes amb molt període, les ones que ens trobem en l'oceà, son molt mes grans, això si, amb menys període, al ser ones mes grans, afavoreix el moviment de l'energy harvesting, i podrem acumular molta mes energia.

11 AMPLIACIÓ AMB MULTIPLEXOR

Un cop complerts els objectius del projecte, s'ha realitzat una petita ampliació per obtenir els resultats amb més facilitat.

Degut a que la boia per dur a terme les proves és troba segellada hermèticament, fer el canvi de condensador a fotosensor per obtenir dades, és una feina bastant pesada, ja que, has de treure tot el segellat i tornar-li a aplicar abans de llençar-la al mar.

Això és degut a que com podem veure durant el capítol del pin out del ESP8266 només tenim un pin de lectura analògica, per aquest motiu hem d'anar variant el cablejat depenent el que volem mesurar.

Un cop extretes les dades s'ha realitzat una programació extra amb un multiplexor, el que podem fer amb el multiplexor és connectar tants aparells com entrades tingui i anar variant la sortida, el qual anirà connectat a l'entrada analògica, és a dir, el multiplexor ens permet escollir quin sensor volem llegir, amb una sola sortida analògica.

El multiplexor escollit ha estat el CD74HC4067, un multiplexor de 16 entrades, per tant, tot i que nosaltres només volem mesurar dos, podríem arribar a connectar 16 sensors diferents.

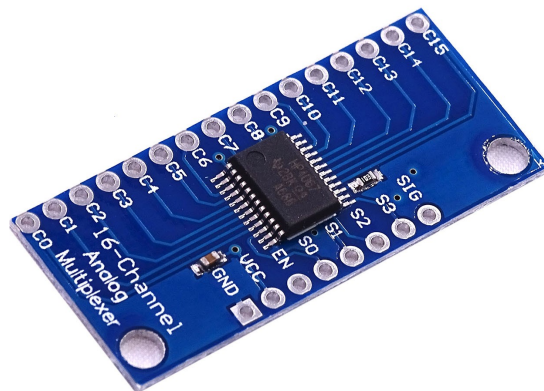


Figura 12.1. Multiplexor CD74HC4067 [<https://leantec.es/tienda/multiplexor-16-canales-cd74hc4067-analog-digital-multiplexer-mux-demux/>].

La taula de la veritat és la següent:

TRUTH TABLE					
S0	S1	S2	S3	\bar{E}	SELECTED CHANNEL
X	X	X	X	1	None
0	0	0	0	0	0
1	0	0	0	0	1
0	1	0	0	0	2
1	1	0	0	0	3
0	0	1	0	0	4
1	0	1	0	0	5
0	1	1	0	0	6
1	1	1	0	0	7
0	0	0	1	0	8
1	0	0	1	0	9
0	1	0	1	0	10
1	1	0	1	0	11
0	0	1	1	0	12
1	0	1	1	0	13
0	1	1	1	0	14
1	1	1	1	0	15

H= High Level
 L= Low Level
 X= Don't Care

Figura 12.1. Taula de la veritat multiplexor [datasheet CD74HC4067]

Per tant, com podem veure, depenent del voltatge que donem a S0, S1, S2 i S3, obtindrem el valor d'una entrada o una d'altre.

En el nostre cas hem utilitzat els canals 0 (no donem voltatge a cap pin, tots es troben a LOW), i el canal número 2, el fet d'escollir aquets dos canals és per dos motius, en primer motiu el canal número 0 és lògic, ja que, no se li ha d'aplicar cap voltatge per obtenir aquesta entrada, d'aquesta manera no generem tant consum, després podíem escollir les entrades 1, 2, 4, 8, que son les entrades que amb la variació d'un pin obtenim el canvi de canal.

Les connexions realitzades han estat en primer lloc la sortida analògica del condensador en l'entrada analògica del multiplexor numero 0, la sortida analògica del fotosensor, en l'entrada del multiplexor número 2 i tots els terres connectats en el mateix pin del ESP8266 juntament amb el pin enable del multiplexor, condició indispensable per que funcioni, hem connectat la S1 a un pin del ESP8266 (LOW – canal0, HIGH – Canal2), i per acabar l'últim que ens queda es alimentar el multiplexor això ho hem fet mitjançant el pin VCC del ESP8266.

Finalment s'ha programat l'entorn LabView, de tal manera que ara la trama que envia Labview al ESP8266 conté un 0 o un 1 mes al final, aquest serà el que decideixi quin és la lectura que volem fer, si enviem un 0, el voltatge per el pin del multiplexor serà LOW, per tant obtindrem la lectura del condensador, de la mateixa manera, al enviar un 1 per LabView, li estem dient al ESP 8266 que enviï voltatge per el pin connectat a la S1 del multiplexor, en aquest cas llegirem el valor de l'entrada número 2, on tenim connectat el fotosensor.

12 DADES DEL PROJECTE

12.1 DESPESA ECONÒMICA

Els elements i materials que s'han comprat per realitzar aquest projecte han estat:

- Bola Transparent: 11,21€.
- Espuma: 6,95€.
- Node ESP: 7,49€.
- TD1205P: Prestat per SARTI.
- Connector bateria 9V: Prestat per SARTI.
- Connector magnètic: 5,95€.
- Pila 9V: 2,17€.
- Claus, cinta aïllant, cablejats: 15€.
- Plom: Prestat per SARTI.
- Cable TTL: 23,95€.

El cost total d'aquest projecte ha estat de 72,72€.

12.2 DURADA DEL PROJECTE

La data de matriculació d'aquest projecte a la universitat politècnica de Vilanova i la Geltrú va ser el dia 28/09/2018, per tant la durada d'aquest projecte ha estat de 8 mesos de durada aproximada.

Durant el primer mes és van dedicar tots els esforços a la captació de teoria sobre el funcionament del Node MCU ESP8266 i a com el TD1205P treia la informació per el port sèrie.

Degut a l'experiència personal amb el projecte de l'any anterior, s'ha anat redactant a mesura que anaven avançant els mesos, per d'aquesta manera, no haver de fer un sprint final redactant tota la informació.

Al llarg dels 3 mesos següents, tota la dedicació és va centrar en la programació, els dos primers, van ser amb el Tera Term i el node MCU, per la implementació del port fictici, per poder veure com rebíem la informació. Un cop vam veure com rebíem la informació, els següent mes es va dedicar a la connexió UDP.

Degut als pocs coneixements amb el programa Labview, aquest punt va ser un dels que mes es va allargar, treballant en paral·lel amb la programació del node, contant amb setmanes d'estudi per aprendre el funcionament del programa, va durar quasi tres mesos la programació i la implementació del codi, fins que finalment funcionava correctament.

Finalment, els dos-tres últims mesos es van destinar a la programació d'Ubidots. Juntament amb la recerca d'un programa més eficaç (sense cap resultat) es va fabricar la boia, es van fer les proves i els assajos a casa, al Centre tecnològic de la UPC de Vilanova i la Geltrú i als realitzats al mar.

13 TREBALLS FUTURS

En el projecte de l'any anterior, vam establir com a possible treball futur, l'anàlisi del moviment de la boia respecte el moviment de les corrents del mar, per poder preveure el moviment del nostre mecanisme. Com vam explicar aquest estudi depèn de molts factors, i matemàticament, és complicadíssim de realitzar.

Amb aquest projecte, podem obtenir moltes més dades del moviment de l'energy harvesting, un possible treball, el qual faria en cas d'haver de fer un tercer, seria construir un altre boia exactament igual, mateixes dimensions, mateixos materials i exactament el mateix pes, però en aquest, sense l'energy harvesting en el interior, per d'aquesta manera poder anar al mar i obtenir els resultats de les dos boies en exactament les mateixes condicions.

La finalitat d'aquest estudi seria poder veure com afecta el moviment de l'energy harvesting en la boia, no només pel pes introduït, sinó per l'absorció d'energia, degut als moviments dels pènduls del seu interior alterant el moviment de la boia.

CONCLUSIONS

Durant el transcurs d'aquest projecte, és va establir com a objectiu l'anàlisi i l'estudi d'un captador d'energia en una boia WAVY drifter.

Al concloure el projecte, podem veure que els aparells programats funcionen correctament, i que l'anàlisi dels resultats han estat molt bons.

Tot i que s'esperaven uns resultats mes elevats, els dies en que és va anar a fer el test al mar, no eren els mes adequats, ja que, el nivell de les ones no era suficient en cap dels dies.

També cal recalcar que les proves és van dur a terme prop de la costa, en aquestes ubicacions solem trobar ones amb bastant període però de petites magnituds, si ens endinsem en l'oceà, trobem ones molt mes grans, però amb menys període.

Un tema pendent que ens ha quedat a l'hora de dur a terme el projecte és l'entorn UBIDOTS. Aquesta aplicació WEB no és troba preparada per pujar dades a alta velocitat, ens serviria a l'hora de penjar dades tals com ubicació, o altres que no hi hagi molt trànsit, doncs a l'hora de pujar les acceleracions, tardava aproximadament 3 segons, el TD1205P, continuava enviant dades al buffer del port sèrie i això feia que es col·lapses. Per aquest motiu, s'ha deixat la programació, per si mes endavant és vol penjar la GEO localització.

Com a experiència personal ha estat un projecte molt interessant, amb moltes hores de dedicació però a l'hora amb molts coneixements assolits durant el seu transcurs.

Com a conclusió final queda l'espina de no haver pogut provar la boia un dia que hi haguessin ones de major magnitud, o anar a alta mar i veure el seu comportament, tot i així, a nivell personal moltíssima satisfacció amb el resultat final.

AGRAIMENTS

No m'agradaria concloure aquest projecte sense abans mostrar l'agraïment pel suport, l'ajut i la dedicació en el transcurs de tots aquests mesos de Matias Carandell, investigador i col·laborador del grup de recerca SARTI i del tutor d'aquest projecte i professor de la Universitat Politècnica de Vilanova i la Geltrú Joaquim del Rio.

BIBLIOGRAFIA

Treballs acadèmics

[1] Cardona Díaz, Laura. Estudio de la corriente superficial en Canarias mediante el seguimiento de boyas de deriva. Tesis Doctoral. Dpto. Física. Universidad de las Palmas de Gran Canarias. 2015.

[2] Carandell, Matias. Tuning the power consumption of a wireless sensor network for sea water monitoring

[3] Carandell, Matias. Design a developement of a kinetic energy harvesterfor oceànic drifter applications.

[4] Valls, Lluís. Estudi d'un captador d'energia electromagnètic a partir de les vibracions generades per l'onatge incident en una boia.

Llibres

[5] Cory L. Clark, LabView Digital Signal Processing and Digital Communications, McGraw-Hill

[6] Thomas J. Bress, Effective LabView Programming

Webgrafia

[7] www.luisllamas.es

[8] <https://hetpro-store.com/TUTORIALES/puerto-serial/>

[9] <https://www.elprocus.com/basics-of-uart-communication-block-diagram-applications/>

[10] <https://lastminuteengineers.com/esp8266-nodemcu-arduino-tutorial/>

[11] [http://www.esploradores.com/comparacion-de-placas-nodemcu /](http://www.esploradores.com/comparacion-de-placas-nodemcu/)

[12] <https://arduino.cc>